ITERATIVE DECODING COMBINED WITH PHYSICAL-LAYER NETWORK CODING ON IMPULSIVE NOISE CHANNELS

Ph.D. Thesis of:

School of Electrical, Electronic and Computer Engineering,

Newcastle University

By

Yuanyi Zhao

Under the Supervision of

Dr. Charalampos Tsimenidis — Newcastle University

Dr. Martin Johnston — Newcastle University

2016

Abstract

This thesis investigates the performance of a two-way wireless relay channel (TWRC) employing physical layer network coding (PNC) combined with binary and non-binary error-correcting codes on additive impulsive noise channels. This is a research topic that has received little attention in the research community, but promises to offer very interesting results as well as improved performance over other schemes. The binary channel coding schemes include convolutional codes, turbo codes and trellis bit-interleaved coded modulation with iterative decoding (BICM-ID). Convolutional codes and turbo codes defined in finite fields are also covered due to non-binary channel coding schemes, which is a sparse research area. The impulsive noise channel is based on the well-known Gaussian Mixture Model, which has a mixture constant denoted by α. The performance of PNC combined with the different coding schemes are evaluated with simulation results and verified through the derivation of union bounds for the theoretical bit-error rate (BER). The analyses of the binary iterative codes are presented in the form of extrinsic information transfer (ExIT) charts, which show the behaviour of the iterative decoding algorithms at the relay of a TWRC employing PNC and also the signal-to-noise ratios (SNRs) when the performance converges. It is observed that the non-binary coding schemes outperform the binary coding schemes at low SNRs and then converge at higher SNRs. The coding gain at low SNRs become more significant as the level of impulsiveness increases. It is also observed that the error floor due to the impulsive noise is consistently lower for non-binary codes. There is still great scope for further research into non-binary codes and PNC on different channels, but the results in this thesis have shown that these codes can achieve significant coding gains over binary codes for wireless networks employing PNC, particularly when the channels are harsh.

Table of Contents

List of Tables

List of Abbreviations and Acronyms

APP *a posteriori* probability (APP)

BER bit error rate (BER)

BICM Bit Interleaved Coded Modulation

EM electromagnetic (EM)

ExIT chart extrinsic information transfer chart (ExIT) chart

GF(q) Galois Field with prime number of q

GMM Gaussian Mixture Model

LLR Log-likelihood ratio (LLR)

MAP Maximum A posteriori Decoding Algorithms (MAP)

NSC Non-Systematic convolutional (NSC)

pdf Probability Density Function

PNC Physical-layer Network Coding

QPSK Quadrature Phase Shift Keying (QPSK)

RSC recursive systematic convolutional (RSC)

RSC Recursive Systematic Convolutional (RSC)

SISO soft-in-soft-out (SISO)

SNR signal-to-noise ratio (SNR)

TWRC two-way relay channel (TWRC)

Chapter 1

Introduction

CHAPTER I

Physical layer network coding (PNC) is an extension of network coding to the physical layer that exploits the addition of electromagnetic signals summed at a relay of a two-user wireless network. The summed electromagnetic signals are demapped to soft values that represent the exclusive-OR (XOR) of the binary messages of two users. When this is broadcast back to the users, the desired message can be obtained by performing the XOR operation on the summed binary message and the user's original binary message. Hence, both users have exchanged messages in a fast and efficient way, improving the throughput of the network. However, it is well known that there is a degradation in performance due to the interference at the relay and it is important to address this. Therefore, the combination of PNC with binary and non-binary error-correcting codes will be investigated in order to improve performance.

This thesis investigates the performance of a two-way wireless relay channel (TWRC) employing PNC at the relay combined with different coding schemes on additive impulsive noise channels. The coding schemes are turbo codes, trellis bit-interleaved coded modulation with iterative decoding (BICM-ID) and non-binary convolutional codes and turbo codes defined in different Galois Fields with a cardinality of q (GF(q)). The type of PNC used throughout the thesis is called link-by-link PNC, where decoding and encoding takes place at the relay as well as at the source nodes.

1.1 Motivation and Challenges

Physical layer network coding has been a very active research area over the last 10 years. It is a technique that can significantly increase the throughput of a wireless network, but at the cost of degrading the overall performance. The literature contains many papers investigating error-correcting codes combined with PNC to improve the performance on the additive white Gaussian noise (AWGN) and fading channels. However, there are two areas that do not seem to have been considered in the literature, but could provide very

interesting results for the academic community. First, the performance of PNC on wireless networks when the channels suffer from impulsive noise has not been investigated. In wireless environments, impulsive noise is a result of man-made interference that introduces very large noise samples that occur for very short time periods into the transmitted signal. This means that the overall noise at a receiver has a non-Gaussian distribution and there are several methods to model impulsive noise, such as the Gaussian mixture model (GMM), symmetric alpha-stable distributions, Middleton class A noise and Bernoulli-Gaussian noise. It is likely that a wireless network could be subjected to impulsive noise and it is important to be able to understand how it would affect the performance of PNC, which this thesis addresses. Second, all papers on PNC focus solely on binary error-correcting codes, but it appears that none consider non-binary error-correcting codes. It is well known that non-binary codes have a performance advantage over binary codes when a channel contains burst errors, so it interesting to observe if these codes can reverse the degradation in performance due to the interference at the relay due to the summed electromagnetic signals. Furthermore, there appear to be no papers investigating non-binary codes on impulsive noise channels, so it will be very interesting and novel to evaluate the performance of PNC combined with non-binary codes on additive impulsive noise channels.

1.2 Aims and objectives

The aim of this thesis is to investigate the performance of a TWRC employing PNC at the relay combined with binary and non-binary error-correcting codes on additive impulsive noise channels. Simulation results of PNC combined with binary turbo codes and trellis BICM-ID will be presented and compared with PNC combined with non-binary convolutional codes and turbo codes, on impulsive noise channels with different levels of impulsiveness. Theoretical analyses will also be provided, with extrinsic information transfer (ExIT) charts to show the behaviour and convergence of the iterative decoding schemes at the relay and also bounds on the bit-error rate (BER) performance of these coding schemes will be derived to validate the simulation results.

The objectives of this project are:

- To investigate the design criteria of convolutional codes, turbo codes and trellis BICM-ID on the AWGN channel and impulsive noise channels.
- To derive theoretical performance bounds for coded PNC systems and design suitable codes to optimize performance.
- Develop ExIT chart analyses to compare the capability of different iterative codes.
- To investigate the design criteria of convolutional codes and turbo codes over GF(q) on PNC over impulsive noise channel.

1.3 Statement of Originality

The accompanying thesis submitted for the degree of Doctor of Philosophy is entitled 'Iterative Decoding Combined with Physical-Layer Network Coding On Impulsive Noise Channel'. This thesis is based on the work conducted by the author in the Department of Electrical and Electronic Engineering, University of Newcastle during the period between April 2012 and July 2016. All the work recorded in this thesis is original unless otherwise acknowledged in the text or by references. This work has not been submitted for another degree in this or any other University.

Signed.....Date.....

1.4 Organisation of the thesis

Chapter 2 presents a literature review covering wireless networks employing PNC, different channel models and coding schemes including convolutional codes, turbo code and trellis BICM-ID used in this thesis.

Chapter 3 begins with a theoretical background on PNC, including the way PNC exchanges information using a simple channel model, which is followed by the description of the AWGN channel and Gaussian mixture model. Also, an introduction to binary convolutional codes is given, explaining the encoding and decoding processes, along with a theoretical performance analysis, which will be used when investigating the more advanced coding schemes in the later chapters. This chapter ends with a summary of finite fields in order to understand the arithmetic required in the encoding and decoding of non-binary codes.

Chapter 4 including the construction of binary turbo coded PNC and the iterative decoding process of turbo codes at the relay are described. Also the theoretical performance analysis of turbo codes is given, by deriving upper bounds on BER performance and performing an ExIT chart analysis to observe the behaviour of the iterative decoder used in binary turbo codes and determine the pinch-off SNR where the decoder converges. Next, trellis BICM-ID is detailed with explanations of the encoder and iterative demapper/decoder and how it is combined with PNC. ExIT charts are also provided to verify the pinch-off SNR of trellis BICM-ID demapper/decoder. Finally, simulation results are presented for PNC combined with turbo codes and trellis BICM-ID on different impulsive noise channels and are compared.

Chapter 5 introduces non-binary convolutional codes defined in finite fields and explains the encoding and decoding processes. A description of how PNC is combined with non-binary convolutional codes is given and simulation results are presented on different impulsive noise channels and compared with binary convolutional codes. A bound on the BER performance of PNC combined with non-binary convolutional codes on impulsive noise channels is also presented to validate the simulation results.

Chapter 6 explains non-binary turbo codes, including the encoding and decoding processes and how they are combined with PNC. Since there is very little information on non-binary turbo codes in the literature, a numerical worked example is provided for one

complete turbo decoder iteration. Simulation results for PNC combined with a non-binary turbo code defined in GF(4) are presented and compared with PNC combined with a binary turbo code on impulsive noise channels. Lower bounds on the BER performance are also derived to show the error floor region of non-binary turbo codes defined in GF(4) and GF(16) and these are also compared with the error floor region of binary turbo codes.

Finally, chapter 7 concludes the work presented in this thesis and provides suggestions for future work.

1.5 Publications arising from this research

1. Zhao, Yuanyi, et al. "Link-by-Link Coded Physical Layer Network Coding on Impulsive Noise Channels." *Sensor Signal Processing for Defence (SSPD), 2015*. IEEE, 2015.

2. Zhao, Yuanyi, et al. "Convolutional Codes Defined In GF(q) Combine With PNC Over Impulsive Noise Channel." *IET journal.2016*

3. Zhao, Yuanyi, et al. "Non-Binary Turbo Coded Physical Layer Network Coding on Noise Channels." *Electronics letters. 2016*

Chapter 2

Literature Survey

2.1 Introduction

In the modern wireless communication area, one of the biggest challenges is to eliminate the interference at the receiver, when the signals are received from multiple sources simultaneously. As a matter of fact, most solutions are proposed from the view of reducing or avoid the interference by the design of a proper receiver to schedule the transmission schemes [1]. However, with the scheme of physical-layer network coding (PNC), it is possible to embrace the interference in order to improve the throughput performance, but there are two aims that must be met: First, the simultaneously received signals at the relay must be able to be interpretable and broadcast from the relay to the destination nodes. Secondly, the destination node must be able to extract the information from the received signals sent from the relay. The capability of network coding schemes provides a potential approach to meet these aims though simple Galois field additions [2][3]. The concept of PNC is to deal with signal reception and modulation through the relay at the physical layer so that the EM signals are mapped by the GF(q) addition of bit streams so that the interference can be used as part of the arithmetic operation in the PNC. Thus, we are curious to find out the performance of PNC with different channel coding schemes over a variety of noise channels, in order to see the capability of PNC and its possible applications.

2.2 Physical-layer Network Coding

The concept of PNC was first proposed in 2006 [4], in order to enhance the performance of wireless network transmission, and since then it has developed into a new field of network coding with wide implications. The basic idea of PNC is to maximize the network coding usage, which occurs naturally when electromagnetic (EM) waves are superimposed on one another. This simple idea turns out to have profound and fundamental ramifications. Since then, many researchers have made contributions in the area of PNC. Afterwards, Zhang introduced the basic concept of synchronization of PNC

in [5], and the importance of the transmitted signals from each node to be synchronized by listing the impact of imperfect synchronization. Katti *et.al* addressed the XOR operation for the received mixing signals from different sources at the relay [6], and evaluates the design on a multiple nodes model. Hausl *et.al* proposed a joint network-channel coding based on turbo codes for multiple-access relay channel [7], which shows the capability of PNC for the cooperative uplink for two mobile stations with the aid of a relay, and shows that systems with network coding for the multiple-access relay channel gain cooperative diversity compared to the system with the distributed turbo code for the relay channel. Popovski proposed the model of a TWRC on PNC, and the application of network coding arising from this model [8]. Partan in [9] has highlighted the important practical issues of PNC.

Zhang proposed another paper in [10] to show the direct application of network coding at PNC for TWRC and subclass the technique into two classes: PNC over a finite field and PNC over an infinite field. Katti *et.al* [11] describes a system that improves the throughput of wireless networks on PNC by mapping the signals on symbol levels, which allows the nodes to opportunistically route groups of bits to their destination with low overhead.

In 2009, Zhang investigates the link-by-link channel coded PNC to show that the performance of Repeat Accumulate (RA) codes on the channel at the nodes in a relay system [12]. This was followed by Rossetto and Zorzi who proposed the design of practical asynchronous PNC design to show the advantage of asynchronous PNC [13]. Koike-Akino *et.al* investigated optimized modulation schemes for TWRC on PNC, showing that QPSK constellations with an XOR network always perform the best for the broadcast stage of PNC [14]. Furthermore, Cui and Gao showed several new transmission schemes for TWRC on PNC with differential modulation schemes.[15]

Further studies by various researchers have led to many new outcomes in the areas of wireless communication and wireless networking. In [17], Louie proposed an analysis and performance comparison of practical PNC for two-way relay channel (TWRC), which compares the performance of traditional transmission schemes, e.g. four time-slots transmission schemes to the performance of PNC and show that the maximum sum-rate of PNC is higher. Nazer explored the core ideas behind PNC and showed the possibilities PNC offers for communications over interference limited wireless networks [18]. Another performance comparison is shown in [19] by Wilson, proving that PNC can

outperform the recently proposed analog network coding schemes. The benefits of PNC are not only limited by synchronous transmission, but Lu also showed that from the asynchronously transmitted signals of the PNC received at the relay, the performance can be improved by applying a belief propagation method [20].

2.3 Channel Coding and PNC

The introduced research on PNC does not include channel coding, but channel coding schemes are widely applied to PNC in order to improve the performance of PNC. Thus, it is important to learn the application of channel coding schemes combined with PNC.

2.3.1 Convolutional codes on PNC

Elias first proposed the class of binary convolutional codes in 1955 [65], and it is one of the most commonly used channel coding schemes with a vast range of applications. Khan considered a physical-layer simulation with quadrature phase shift keying (QPSK) modulated convolutional code in [26] to serve as a helpful resource for researchers. A.Zhan investigates the advantage of linearity of the channel coding schemes combined with PNC in [27], analysing the capacity of the channel coded model on the AWGN channel in order to show the ability of PNC to improve the network throughput and robustness. Wang discussed the channel coding design on PNC under a three-node network coding scenario in [28], where the bit error rate (BER) at the relay of the channel is bounded by the weakest channel coding scheme. Gacanin presented the performance of bi-directional transmission with convolutional coded PNC on a multipath channel in [29]. D. To has shown that the Viterbi algorithm can be used by approximating the maximum likelihood (ML) decoding for the XORed message at the relay in [30] and by applying the convolutional codes reduced-state decoding can achieve the same diversity gain as full-state decoding for fading channels. The implementations of asynchrony between signals transmitted by the multiple transmitters, which integrate channel coding with PNC to achieve reliable communication, are looked into in [31]-[32]. The asynchrony decoding process of convolutional coded PNC systems is investigated by Yang. Overall, the convolutional codes are treated as the most commonly use channel coding schemes in PNC, and as a fundamental background of turbo codes or other

coding schemes, it is important to look into the performance of convolutional codes combined with PNC.

2.3.2 Turbo Codes combined with PNC

It is widely acknowledged that iterative decoding schemes have significant advantages when combined with PNC. In [33], Hausl introduced an extension of the conventional two-way relay communication with a joint network-channel coding method for PNC, where turbo codes are used as the channel coding scheme for both source nodes and a network code is used at the relay. They derived closed-form expressions for upper bounds on channel capacities of the time-division relay and two-way relay channel without power control. The study of [34] proposed a joint physical network coding with turbo codes for multiple-access channels, where the performance of the proposed scheme approached the information-theoretic limits of the traditional network. They showed that combining channel coding and network coding achieves a 2.1dB improvement at a bit error rate (BER) of 10^{-5} on the AWGN channel over traditional network coding with a turbo multi-user detector. The study of Fang *et al.* [35] investigated the performance degradation of hierarchical decode-and-forward (HDF) turbo coded PNC on conventional two-way relay communications compared to a single user end-to-end turbo coded system. A simple upper bound on the performance of turbo codes using ExIT charts was also presented. Guan [36] showed an improved PNC method based on turbo codes and M-PSK, analyzing the transmission energy consumption of the proposed scheme and showing how the enhanced PNC method can halve transmission energy consumption at the relay node over conventional PNC. Zeng [37] presented the non-coherent detection of iterative differential phase-shift keying (DPSK) demodulation for PNC combined with turbo codes on conventional two-way relay communications, where the iterative processing converges faster on a Rayleigh fading channel, and an ExIT chart analysis shows that most of the coding gains are achieved within two iterations.

2.3.3 Trellis BICM combined with PNC

There are several studies on trellis BICM combined with PNC on conventional two-way relay communications. In [38], Xu *et.al* showed that trellis BICM can significantly improve the BER performance of a PNC system by applying a suitable iterative demapping and decoding framework and proper constellation mapping schemes specially designed for PNC. By considering the deployment of a relay transceiver employing PNC and BICM-ID coding, the performance of the receiver achieved an energy efficiency improvement of 0.5-0.9dB on fading channels. Tao [39] addressed the convergence behavior of the iterative receiver of BICM-ID coded PNC at the relay using (ExIT) charts to design good channel-coded PNC schemes. Noori introduces the concept of semi-Gray mapping on PNC that improves the system BER performance and the achievable rate in [40]. It needs to denotes that based on our system, the anti-Gray mapping could reduce the complexity at the relay, due to anti-Gray mapped signals addition could achieve the coding gain which semi-Gary mapping could not. It turns out research in the area of trellis BICM combined with PNC, which is one of the commonly used iterative coding schemes, is sparse and it is therefore important to investigate the performance of trellis BICM combined with PNC.

2.3.4 Non-Binary Convolutional Codes on PNC

One important class of error-correction code is the convolutional code. The major difference between convolutional code and block codes is that block codes are only able to encode a fixed length of information bits. However, convolutional codes can encode a continuous stream of information bits. Another advantage of the convolutional code is its simplicity: convolutional codes have a much simpler trellis than block codes. However, convolutional codes achieve a poorer performance than block codes at higher code rates. Multiple levels encoding to form q-ary codes was first proposed in [59] and then their performance over different channels was proposed in [60-62]. Also, this class of code is analyzed by [63][64]. It is well known that convolutional codes construct defined in

GF(q) outperform binary convolutional codes, and they have become more popular in recent research [64-68]. So our motivation is to compare the performance of non-binary convolutional codes combined with PNC over GMM channel with different mixtures of impulsiveness to binary convolutional codes, in order to see how the impulsive noise affects different coding schemes and evaluate it by deriving the upper bound on performance.

There are two classes of non-binary convolutional code that provide substantial coding gains when used with M-ary orthogonal signaling and non-coherent detection. The first class is the q-ary convolutional code, which can be viewed as an extension of dual-k codes with much larger memory elements and the symbols are mapped to the field of GF(q). The second class is the binary-to-q-ary convolutional codes, which can be viewed as a rate $\frac{1}{n}$ extension of the rate 1 code [69]. It turns out that the research of non-binary convolutional codes combined PNC is rare. Faraji-Dana investigated a non-binary constellation modulation scheme combined with PNC and convolutional coding in [70], where the simulated FER of the non-binary convolutional code is presented with a upper bound on the performance of decoding the network combinations. However, their works only considered the field of GF(2) to GF(4) with simple comparison to the basic convolutional codes of constraint length two. In this work we are going to consider the scenarios where the non-binary convolutional codes are mapped onto higher order of fields with longer constraint lengths, e.g. the industry standards, to approach a practical performance for non-binary convolutional codes combined with PNC.

2.3.5 Non-binary Turbo Codes on PNC

As one of the most significant breakthroughs in coding, turbo codes soon became one of the most popular coding schemes. However, there is only a small body of work on turbo codes on the TWRC with PNC. This includes Hausl [78] who proposed a distributed turbo coding scheme for a multiple access relay channel and Feng [79] who studied the performance degradation of binary turbo codes on a TWRC. However, there appears to be no work in the literature studying non-binary turbo codes with PNC. In [80] Berrou

introduced non-binary convolutional codes for turbo coding and showed that Quaternary codes can be advantageous, both in terms of performance and complexity. Hence, in this thesis we investigate the performance of non-binary turbo codes over GF(4) and GF(16) combined with PNC, where encoding and decoding take place at the relay and source nodes in each time slot. We also consider the effect of additive impulsive noise on the performance of non-binary turbo codes. This has been investigated in [81], where the authors analyzed the performance of binary turbo codes combined with PNC on additive impulsive noise channels.

2.4 Channel Coding and Impulsive Noise Channel

Although channel coding combined with PNC is becoming a more popular research area, most of the research has been done under the assumption of AWGN channel due to the pdf of Gaussian noise being straightforward for the system to be analyzed. However, it is important to look the performance of PNC over other more complicated channels. One noise model we are particular interested in is the impulsive noise model, as there are many scenarios in a real environment that can be modelled as impulsive noise channel, such as underwater environment and so on. Again, there are few works that focus on the impulsive noise channel with PNC, and none mention channel coded PNC. In [98], Chitre provides an overview of the key developments in communications techniques of underwater networks that discussed the open problems and the challenges of the impulsive channel in the near future. Yousuf discussed the impulsive noise events in the error correction and data interleaving codes with varying code rates in [99], proving that the effects of impulsive noise can be reduced by applying channel coding schemes. Overall, the area of channel coded PNC over impulsive noise channel still needs to be explored, and is one of the aims of this thesis.

2.5 Conclusions

After a literature survey on past research on TWRC with PNC, it appears that most researchers are assuming the network coding system is on AWGN channels and only very few researchers are focusing on other noise channels. Furthermore, the non-binary channel coding schemes combined PNC is essentially and unknown area. We have chosen four different channel coding schemes that are combined with PNC, in order to observe the BER performances on PNC over both the AWGN channel and impulsive noise channels, along with the theoretical performance analysis including the bound theoretical derivation and ExIT charts for iterative decoding codes to verify our results. We believe that the works in this thesis are novel and the results are original, and not only explore the core ideas behind the PNC but also fill in an important gap in the available literature that will be useful for other researchers considering this area.

Chapter 3

Theoretical Background

3. Theoretical Background

3.1 Introduction

In this chapter, the required background knowledge for physical layer network coding (PNC), impulsive noise channel modelling, encoding and decoding of convolutional codes and finite field arithmetic required to understand non-binary codes is presented. The chapter begins by explaining the system model for PNC, where encoding/decoding is performed at the source/destination nodes and additionally encoding and decoding takes place at the relay. From this we can analyse the theoretical and simulated bit-error rate (BER) performance of the two-way wireless relay channel employing PNC at the relay. One of the contributions of this thesis is the addition of impulsive noise to the PNC system model, which is modelled as a Gaussian mixture and defined later in this chapter. Its effect on several different coding schemes, including turbo codes, trellis bit-interleaved coded modulation, non-binary convolutional codes and non-binary turbo codes is evaluated and analysed throughout the thesis. It is therefore important to gain a fundamental understanding of convolutional codes, which are the component codes of the aforementioned coding schemes. Finally, non-binary codes are considered in chapters 5 and 6, which are defined in extension fields, so it essential to include an explanation of finite fields, which concludes this chapter.

3.2 System Channel Model

3.2.1 AWGN model

In wireless communications, the AWGN model is widely used as it provides an accurate description of a complete set of noisy observations by means of qualitative information that is characterized by a few simple parameter values [39]. The well-known AWGN pdf with zero-mean is given by:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x}{2\sigma^2}} \tag{3.1}$$

where σ^2 is the noise variance of the distribution. This distribution plays an important role in statistical modelling and was first derived by German mathematician Johann Gauss in 1867 [47]. As the reference model for this research, the Gaussian distribution acts as one of the main factors in analysing the theoretical performance of each network coding scheme in this thesis. Examples of AWGN noise samples for different SNRs are shown in Fig 3.1.

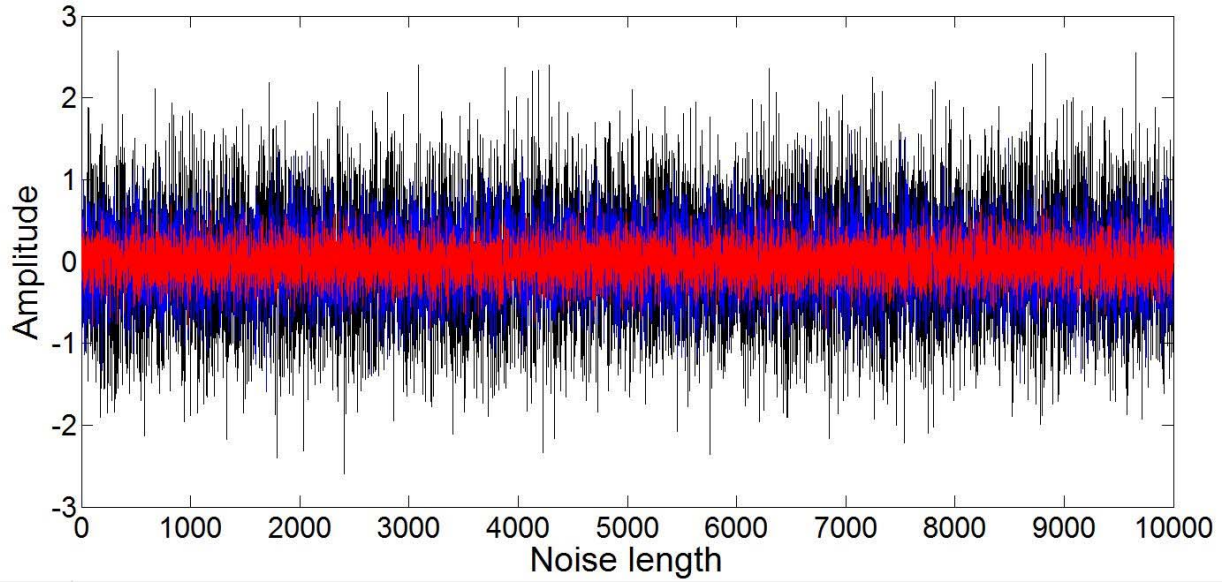


Figure 3.1 Noise strength at different SNR: 0dB (black), 5dB (blue), and 10dB (red).

3.2.2 Impulsive Model

It is not enough to evaluate the performance of network coding only on the AWGN model. Some scenarios can cause impulsive noise to be added during network transmission. In general, there are several different models to represent an impulsive noise environment: the Gaussian mixture model (GMM), symmetric alpha-stable distributions, Middleton class A noise and Bernoulli-Gaussian noise. The mixture model was first purpose by Tukey in 1960 [50], and it is a probabilistic model that assumes the data are generated from a mixture of a finite number of Gaussian distributions, which results in an approximately Gaussian distribution but with a certain number (corresponding to the mixture constant α) of contaminating distributions. Due to the mixture model being convenient to calculate the noise distribution in order to evaluate the network coding performance in an impulsive noise environment, it has widespread popularity among many researchers over the past decade [44-50]. However, the most popular model of the GMM is a mixture of two Gaussian densities. To evaluate the network coding schemes under an impulsive noise environment, it is important to know the distribution of the impulsive model. The pdf of the GMM can be expressed as:

$$p_{GMM} = (1-\alpha)p_G(x) + \alpha p_I(x). \tag{3.2}$$

where $0 \leq \alpha \leq 1$ is the Gaussian mixture constant, with larger values of α denoting more impulsiveness. The terms $p_G(x)$ and $p_I(x)$ are the Gaussian pdf and impulsive pdf respectively, where $p_I(x)$ has a much larger variance than $p_G(x)$, and their addition results in a heavy-tailed distribution p_{GMM}. The noise strength at different α of GMM impulsive noise is shown in Fig 3.2.

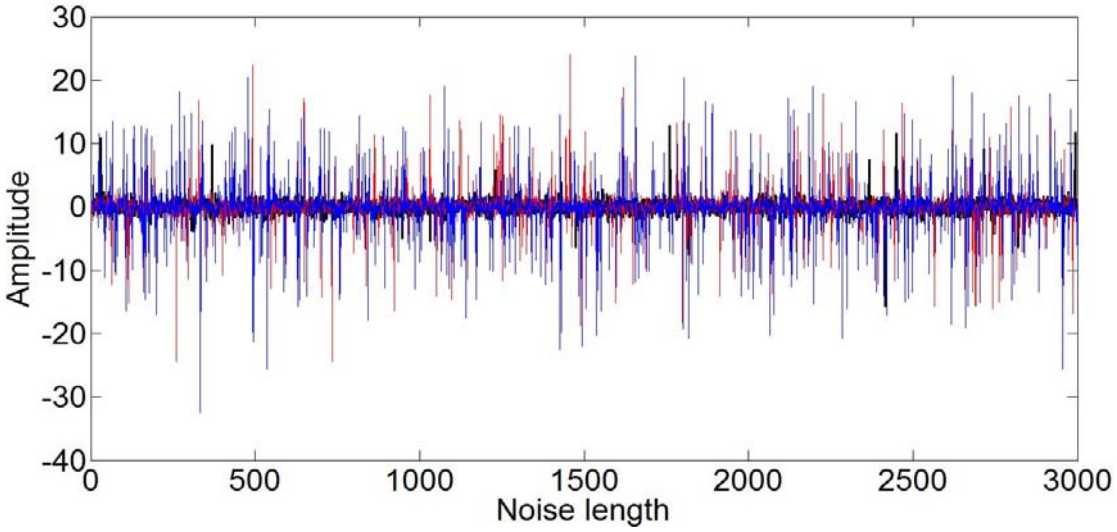


Figure 3.2 Noise strength at various α at SNR=1dB: $\alpha = 0.01$ (black), $\alpha = 0.1$ (red), and $\alpha = 0.5$ (blue).

From Fig 3.2 it can be seen that as the value of mixture constant α increases, the proportion of the distribution $p_I(x)$ increases and thus increases the impulsiveness of p_{GMM}.

In this thesis, we have chosen the variance of impulsive noise σ_I^2 to be ten times larger than the variance of Gaussian noise σ_G^2, where:

$$\sigma_G^2 = \frac{N_0}{2}, \tag{3.3}$$

$$N_0 = \frac{1}{4R10^{\frac{SNR_{dB}}{10}}}. \tag{3.4}$$

We chose the mixture constant α to have values of $0 \leq \alpha \leq 0.5$. For some channel coding schemes, even with one percent of impulsive noise, the performance of the codes can be affected significantly.

Recalling the distribution of GMM in (3.2), the BER of the impulsive channel with BPSK and QPSK modulation is simply defined as:

$$P_{GMM} = (1-\alpha)\frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_G}}\right) + \alpha\frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_I}}\right), \quad (3.5)$$

where N_G and N_i are the noise power spectral densities of the Gaussian noise and impulsive noise respectively.

3.3 Construction of PNC

3.3.1 PNC Transmission Model

Let us consider the following situation: There are two source nodes: node 1 and node 3. Both nodes want to communicate to each other but this is only possible through the aid of a relay as shown in Fig 3.3 [18]：

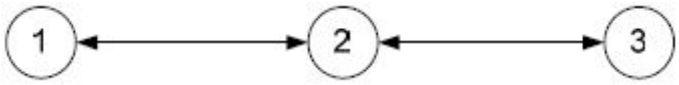


Figure 3.3 Three-node Linear Network.

Fig 3.3 shows a traditional two-way relay channel (TWRC) model, where 1 and 3 are the user nodes and 2 is the relay. In a two-way relay transmission scheme, communication can take place over four-time slots, where node 1 communicates with node 2 in the first two time slots and node 2 remains idle, and node 2 communicates back to node 1 during the last two time slots with node 1 remaining idle, as shown in Fig 3.4 [18].

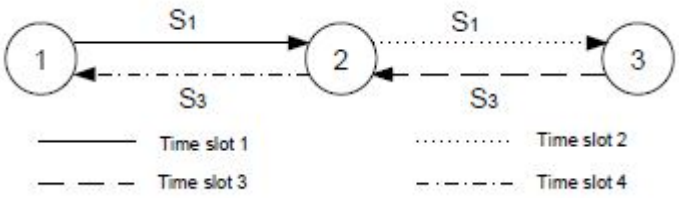


Figure 3.4 Traditional four-time slot transmission scheme.

We refer to this scheme as a four-time slot transmission scheme, and its performance has been studied extensively.. One problem with this transmission scheme is the relatively low throughput, as a consequence of transmission over four-time slots.

One way to increase the throughput is to allow node 1 to transmit S_1 in the first time slot but now let node 3 transmit S_3 in the the second time slot. At the relay (node 2), the signals are added resulting in:

$$S_2 = S_1 \oplus S_3 \tag{3.6}$$

where $\oplus$ denotes the bitwise exclusive OR operation applied to the entire frame of S_1 and S_3. Then node 2 broadcasts the signal S_2 back to both node 1 and node 3 in the third time slot. When the destination receives the message (e.g. node 1), S_3 can be extracted by XOR-ing S_2 and S_1:

$$S_1 \oplus S_2 = S_1 \oplus (S_1 \oplus S_3) = S_3 \tag{3.7}$$

This is known as straightforward network coding and its system model is shown in Fig 3.5 [18]:

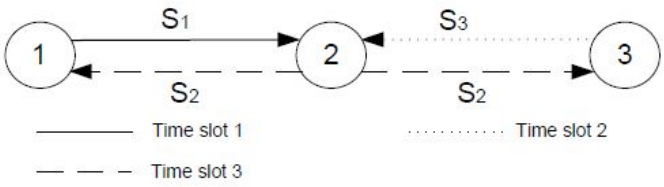


Figure 3.5 straightforward network coding scheme.

Similarly, node 3 can extract information that node 1 sent from the broadcast signal S_2 as well. In this scheme, only three-time slots are needed, meaning the throughput improvement is 33% better than the traditional transmission scheduling scheme.

Although the straightforward network coding scheme boosts the transmitting speed by 33%, it can be improved to 100% with the PNC transmission scheduling scheme, which completes the exchange of signals over two time slots. In the first time slot, node 1 and node 3 transmit to the relay simultaneously, while during the second time slot, the relay forwards the summed received signals to both nodes. Such a scheme is referred to as a

two time slot PNC scheme and has been shown to achieve higher throughput than the four-time slot transmission scheme due to fewer time slots being used for transmission. However, certain assumptions are made to achieve this, e.g. symbol-level and carrier phase synchronization and the use of power control to all nodes so that the signals from node 1 and node 3 received by the relay have the same phase and amplitude. Fig 3.6 shows the PNC system model [18]: The destination nodes perform the same procedure as in the straightforward network coding scheme to extract the information from the signal.

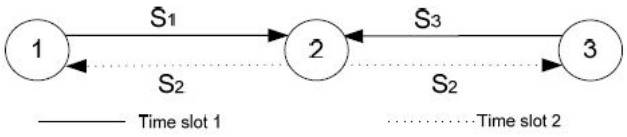


Figure 3.6 PNC system model.

3.3.2 Theoretical Analysis of PNC

For BPSK-modulated PNC, it is possible to derive the theoretical performance for both the relay and the node. Let us consider the channel noise as AWGN, so the error probability is [36]:

$$P\left(r|x=\sqrt{E_b}\right)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{\left(r-\sqrt{E_b}\right)^2}{2\sigma^2}}, \tag{3.8}$$

This gives us the probability of getting an error event when the node is transmitting a $\sqrt{E_b}$. So, the error probability of node 1 transmitting to node 2 without a relay is:

$$P_{single}=\frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_0}}\right), \tag{3.9}$$

where $\frac{E_b}{N_0}$ is the SNR. The received added signals are resulting in three values with four different situations, which are:

Node 1	Node 3	Relay
$\sqrt{E_b}$	$\sqrt{E_b}$	$2\sqrt{E_b}$
$\sqrt{E_b}$	$-\sqrt{E_b}$	0
$-\sqrt{E_b}$	$\sqrt{E_b}$	0
$-\sqrt{E_b}$	$-\sqrt{E_b}$	$-2\sqrt{E_b}$

Table 3.1: Possible values of BPSK modulated signals received at the relay on PNC

From Table 3.1, it can be seen that the probability of receiving a 0 is 50%, and the other two values are 25% respectively. The probability of a bit error at the relay given that the summed BPSK signals equal $-2\sqrt{E_b}$ is

$$P\left(r|x=-2\sqrt{E_b}\right)=\frac{1}{\sqrt{2\pi\sigma^2}}\int_{\sqrt{E_b}}^{-\sqrt{E_b}}e^{-\frac{(r+2\sqrt{E_b})^2}{2\sigma^2}}dr. \tag{3.10}$$

For the probability of a bit error, given that x= +$2\sqrt{E_b}$ we have:

$$P\left(r|x=+2\sqrt{E_b}\right)=\frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\sqrt{E_b}}^{\sqrt{E_b}}e^{-\frac{(r-2\sqrt{E_b})^2}{2\sigma^2}}dr. \tag{3.11}$$

For the probability of a bit error, given that x= 0 we have:

$$P(r|x=0)=\frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{-\sqrt{E_b}}e^{-\frac{r^2}{2\sigma^2}}dr+\frac{1}{\sqrt{2\pi\sigma^2}}\int_{\sqrt{E_b}}^{\infty}e^{-\frac{r^2}{2\sigma^2}}dr. \tag{3.12}$$

The complementary error function is defined as [37]:

$$erfc(x)=\frac{2}{\sqrt{\pi}}\int_{x}^{\infty}e^{-t^2}dt \tag{3.13}$$

So, by applying Eqn. (3.13) to Eqn. (3.10) - (3.12) we obtain:

$$P\left(r|x=-2\sqrt{E_b}\right)=\frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)-\frac{1}{2}erfc\left(3\sqrt{\frac{E_b}{N_0}}\right), \tag{3.14}$$

$$P\left(r|x=+2\sqrt{E_b}\right)=\frac{1}{2}erfc\left(3\sqrt{\frac{E_b}{N_0}}\right)-\frac{1}{2}erfc\left(-\sqrt{\frac{E_b}{N_0}}\right), \tag{3.15}$$

$$P(r|x=0)=\ erfc\left(\sqrt{\frac{E_b}{N_0}}\right). \tag{3.16}$$

By following the probabilities from Table 3.1, the theoretical bit error probability P_{relay} at the relay can be derived as:

$$P_{relay} = 0.25\ P\left(r|x=-2\sqrt{E_b}\right) + 0.25\ P\left(r|x=+2\sqrt{E_b}\right)+0.5P(r|x=0)$$

$$\begin{aligned}= 0.25&\left[\frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)-\frac{1}{2}erfc\left(3\sqrt{\frac{E_b}{N_0}}\right)\right]\\ &+0.25\left[\frac{1}{2}erfc\left(3\sqrt{\frac{E_b}{N_0}}\right)-\frac{1}{2}erfc\left(-\sqrt{\frac{E_b}{N_0}}\right)\right]\\ &+0.5erfc\left(\sqrt{\frac{E_b}{N_0}}\right).\end{aligned} \tag{3.17}$$

Due to terms $erfc\left(\pm3\sqrt{\frac{E_b}{N_0}}\right)$ only having a significant effect on P_{relay} at very low SNRs, we can ignore them to obtain:

$$P_{relay}\approx\ \frac{3}{4}erfc\left(\sqrt{\frac{E_b}{N_0}}\right). \tag{3.18}$$

To derive the theoretical performance at the destination nodes, two situations need to be considered when an error is received at a destination node: an error occurred at the relay during the first time slot and is received unchanged by node 3, or no error occurred at the relay but an error then occured at the destination node during the second slot.

If a bit error occurs at the relay and also at the destination node, then in reality the bit has been flipped twice and no error has actually occured. According to the situations above, the error probability of transmitting signals from source node to destination can be expressed as:

$$P_{destination\_node} = (1 - P_{single})P_{relay} + (1 - P_{relay})P_{single} - 2P_{single}P_{relay}$$

$$= \left(1 - \frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)\right)\frac{3}{4}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)$$

$$+ \left(1 - \frac{3}{4}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)\right)\frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_0}}\right) \qquad (3.19)$$

$$- 2\left[\frac{3}{4}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)\frac{1}{2}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)\right] = \frac{5}{4}erfc\left(\sqrt{\frac{E_b}{N_0}}\right) - \frac{9}{4}erfc\left(\sqrt{\frac{E_b}{N_0}}\right)^2$$

3.3.3 Performance of QPSK modulated PNC at the Relay and Destination Nodes

Figure 3.5 shows the system model of a PNC system from the source nodes to the relay.

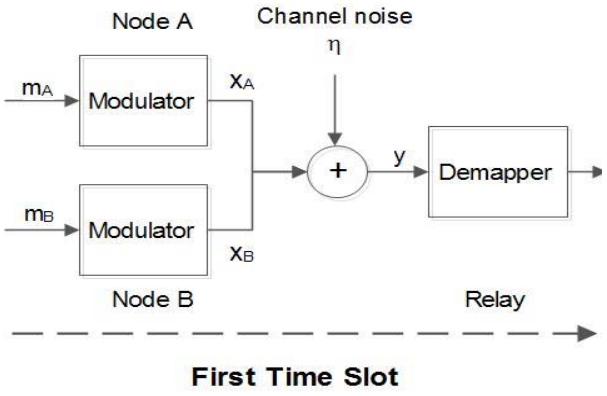


Figure 3.7 The system model evaluating the BER performance of uncoded QPSK PNC at the relay.

It can be seen that node A and node B map the information sequences m_A and m_B to modulated signals x_A and x_B and then transmit these signals to the relay simultaneously. x_A and x_B are added at the relay, which can be expressed as:

$$y = x_A + x_B + \eta, \tag{3.20}$$

where η is the channel noise. Constellation diagrams showing the constellation points at the source node and at the relay without noise are shown in Fig 3.8. Then the theoretical BER performance at the relay can be determined from the nine constellation points.

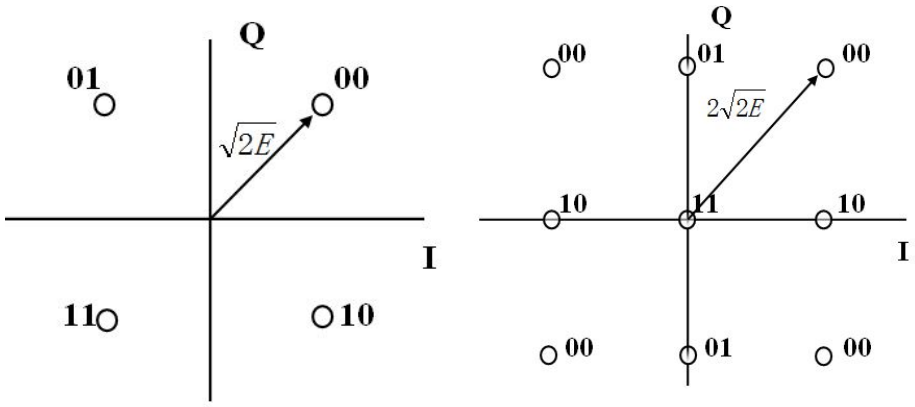


Figure 3.8 Constellation diagrams of the QPSK modulation scheme at the source nodes (left) and the 9-point constellation at the relay (right).

Let us consider the channel noise as AWGN first, where the probability distribution function (pdf) of receiving a signal at the relay given that we transmitted two signals from the source nodes is [36]:

$$P(r|x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left[\frac{(r_I - x_I)^2 + (r_Q - x_Q)^2}{2\sigma^2}\right]}, \tag{3.21}$$

where x is the summed received symbol from the source nodes, x_I and x_Q are real and imaginary part of x, r is the received symbol at the relay and r_I and r_Q are the real and imaginary parts of r.

As shown in Fig 3.8, the received signals when added result in nine different complex values from sixteen possible summations, presented in Table 3.2:

Node 1	x_1	Node 3	x_3	$x = x_1 + x_3$	Relay
00	$\sqrt{E_b} + j\sqrt{E_b}$	00	$\sqrt{E_b} + j\sqrt{E_b}$	$2\sqrt{E_b} + 2j\sqrt{E_b}$	00
00	$\sqrt{E_b} + j\sqrt{E_b}$	01	$-\sqrt{E_b} + j\sqrt{E_b}$	$2j\sqrt{E_b}$	01
00	$\sqrt{E_b} + j\sqrt{E_b}$	10	$\sqrt{E_b} - j\sqrt{E_b}$	$2\sqrt{E_b}$	10
00	$\sqrt{E_b} + j\sqrt{E_b}$	11	$-\sqrt{E_b} - j\sqrt{E_b}$	0	11
01	$-\sqrt{E_b} + j\sqrt{E_b}$	00	$\sqrt{E_b} + j\sqrt{E_b}$	$2j\sqrt{E_b}$	01
01	$-\sqrt{E_b} + j\sqrt{E_b}$	01	$-\sqrt{E_b} + j\sqrt{E_b}$	$-2\sqrt{E_b} + 2j\sqrt{E_b}$	00
01	$-\sqrt{E_b} + j\sqrt{E_b}$	10	$\sqrt{E_b} - j\sqrt{E_b}$	0	11
01	$-\sqrt{E_b} + j\sqrt{E_b}$	11	$-\sqrt{E_b} - j\sqrt{E_b}$	$-2\sqrt{E_b}$	10
10	$\sqrt{E_b} - j\sqrt{E_b}$	00	$\sqrt{E_b} + j\sqrt{E_b}$	$2\sqrt{E_b}$	10
10	$\sqrt{E_b} - j\sqrt{E_b}$	01	$-\sqrt{E_b} + j\sqrt{E_b}$	0	11
10	$\sqrt{E_b} - j\sqrt{E_b}$	10	$\sqrt{E_b} - j\sqrt{E_b}$	$2\sqrt{E_b} - 2j\sqrt{E_b}$	00
10	$\sqrt{E_b} - j\sqrt{E_b}$	11	$-\sqrt{E_b} - j\sqrt{E_b}$	$-2j\sqrt{E_b}$	01
11	$-\sqrt{E_b} - j\sqrt{E_b}$	00	$\sqrt{E_b} + j\sqrt{E_b}$	0	11
11	$-\sqrt{E_b} - j\sqrt{E_b}$	01	$-\sqrt{E_b} + j\sqrt{E_b}$	$-2\sqrt{E_b}$	10
11	$-\sqrt{E_b} - j\sqrt{E_b}$	10	$\sqrt{E_b} - j\sqrt{E_b}$	$-2j\sqrt{E_b}$	01
11	$-\sqrt{E_b} - j\sqrt{E_b}$	11	$-\sqrt{E_b} - j\sqrt{E_b}$	$-2\sqrt{E_b} - 2j\sqrt{E_b}$	00

Table 3.2: Possible values of QPSK modulated signals received at the relay on PNC

From Table 3.2, it can be seen that there are four cases of receiving 00, i.e. the points $2\sqrt{E_b}+2j\sqrt{E_b},\ 2\sqrt{E_b}-2j\sqrt{E_b}, -2\sqrt{E_b}+2j\sqrt{E_b}, -2\sqrt{E_b}-2j\sqrt{E_b}$, which each have the probability of 1/16=6.25%. The probability for each of the points $-2\sqrt{E_b}$, $2\sqrt{E_b}, -2j\sqrt{E_b},\ 2j\sqrt{E_b}$ is 2/16 = 12.5% and the probability of the point 0 is 4/16 = 25%. Thus, the conditional probabilities of the received signal at the relay are:

$$P(y|m_1=0,m_3=0)=$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(e^{-\left[\frac{\left(y_I-2\sqrt{E_b}\right)^2+\left(y_Q-2\sqrt{E_b}\right)^2}{2\sigma^2}\right]}+e^{-\left[\frac{\left(y_I+2\sqrt{E_b}\right)^2+\left(y_Q-2\sqrt{E_b}\right)^2}{2\sigma^2}\right]}\right.$$

$$+e^{-\left[\frac{\left(y_I-2\sqrt{E_b}\right)^2+\left(y_Q+2\sqrt{E_b}\right)^2}{2\sigma^2}\right]}$$

$$\left.+e^{-\left[\frac{\left(y_I+2\sqrt{E_b}\right)^2+\left(y_Q+2\sqrt{E_b}\right)^2}{2\sigma^2}\right]}\right)dy_I dy_Q, \qquad (3.22)$$

$$P(y|m_1=0,m_3=1)=$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(2e^{-\left[\frac{{y_I}^2+\left(y_Q-2\sqrt{E_b}\right)^2}{2\sigma^2}\right]}+2e^{-\left[\frac{{y_I}^2+\left(y_Q+2\sqrt{E_b}\right)^2}{2\sigma^2}\right]}\right)dy_I dy_Q, \quad (3.23)$$

$$P(y|m_1=1,m_3=0)=$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(2e^{-\left[\frac{\left(y_I-2\sqrt{E_b}\right)^2+{y_Q}^2}{2\sigma^2}\right]}+2e^{-\left[\frac{\left(y_I+2\sqrt{E_b}\right)^2+{y_Q}^2}{2\sigma^2}\right]}\right)dy_I dy_Q, \qquad (3.24)$$

$$P(y|m_1=1,m_3=1)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}4e^{-\left[\frac{{y_I}^2+{y_Q}^2}{2\sigma^2}\right]}dy_I dy_Q \qquad (3.25)$$

As we know, the error probability of QPSK is equal to the error probability of BPSK. Thus the error probability at the relay can be simply viewed as:

$$P_{qr} = P_{relay} = \frac{1}{4}P(y|m_1 = 0, m_3 = 0) + \frac{1}{4}P(y|m_1 = 0, m_3 = 1) + \frac{1}{4}P(y|m_1 = 1, m_3 = 0) + \frac{1}{4}P(y|m_1 = 1, m_3 = 1) \quad (3.26)$$

A system model showing both time slots of the TWRC employing PNC is shown in Fig 3.9.

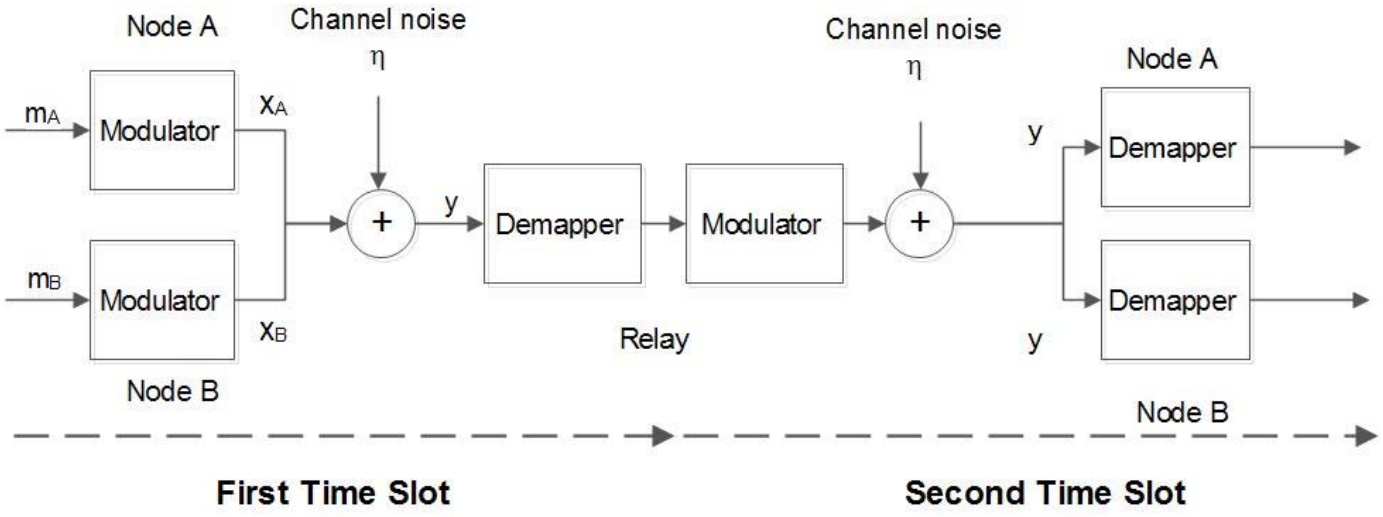


Figure 3.9 System model of evaluating BER performance of uncoded QPSK PNC at destination nodes.

To derive the theoretical BER at the destination nodes, there are two scenarios that need to be considered: First, a bit error is received at the destination node due to noise added at node 1. Second, a bit error occurs at the relay during the second slot and is broadcast to the destinatio nodes where is remains as a bit error. One situation needs to be highlighted: If a bit error occurs at the relay and at the destination node, then the bit has actually been flipped twice and there is no error at the destination node. According to the scenarios above, the probability of a bit error at the destination nodes is:

$$P_{qn} = (1 - P_{single})P_{qr} + (1 - P_{qr})P_{single} - 2P_{single}P_{qr} \ . \qquad (3.27)$$

The comparison between simulated BER and the theoretical BER both at the relay and the destination nodes are shown in Fig 3.10.

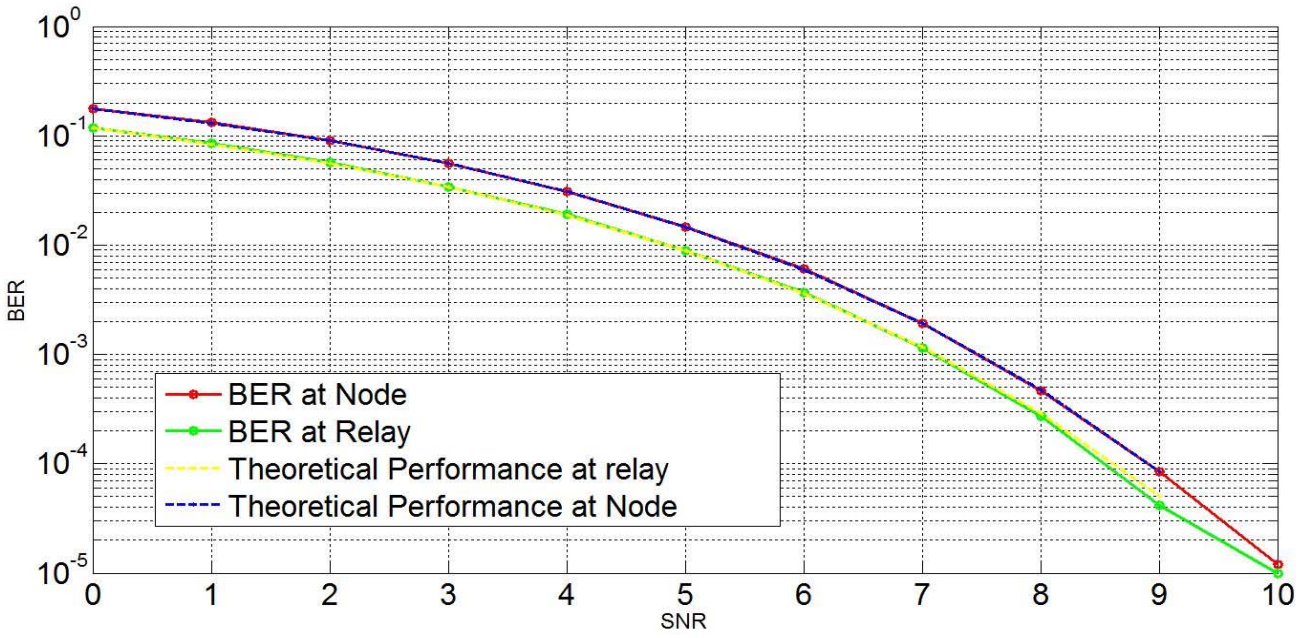


Figure 3.10 Comparison of simulated BER to theoretical BER, P_{qn} and P_{qr} at the relay/node on uncoded PNC.

From Fig 3.10 it can be seen that the simulation results match closely with the theoretical error probability. The performance of decoding at the relay on PNC (red curve) is slightly better than decoding at the destination node (green curve), with approximately a 0.5dB advantage. The reason for this degradation is because if the relay only broadcasts signals without error correction, the signals are interfered by the channel noise twice. Thus, the performance of the BER performance at the relay is very critical, and it is important to find the appropriate network coding scheme for the relay.

By following the constellation diagram in Fig 3.8, the conditional probabilities of received signals at the node are:

$$P_{GMM}(y|x=00) =$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left((1-\alpha)e^{-\left[\frac{\left(y_I-\sqrt{E_b}\right)^2+\left(y_Q-\sqrt{E_b}\right)^2}{2\sigma_G^2}\right]}+\alpha e^{-\left[\frac{\left(y_I-\sqrt{E_b}\right)^2+\left(y_Q-\sqrt{E_b}\right)^2}{2\sigma_I^2}\right]}\right)dy_I dy_Q, \quad (3.28)$$

$$P_{GMM}(y|x=01) =$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left((1-\alpha)e^{-\left[\frac{\left(y_I-\sqrt{E_b}\right)^2+\left(y_Q+\sqrt{E_b}\right)^2}{2\sigma_G^2}\right]}+\alpha e^{-\left[\frac{\left(y_I-\sqrt{E_b}\right)^2+\left(y_Q+\sqrt{E_b}\right)^2}{2\sigma_I^2}\right]}\right)dy_I dy_Q\,, \quad (3.29)$$

$$P_{GMM}(y|x=10) =$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left((1-\alpha)e^{-\left[\frac{\left(y_I+\sqrt{E_b}\right)^2+\left(y_Q-\sqrt{E_b}\right)^2}{2\sigma_G^2}\right]}+\alpha e^{-\left[\frac{\left(y_I+\sqrt{E_b}\right)^2+\left(y_Q-\sqrt{E_b}\right)^2}{2\sigma_I^2}\right]}\right)dy_I dy_Q\,, \quad (3.30)$$

$$P_{GMM}(y|x=11) =$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left((1-\alpha)e^{-\left[\frac{\left(y_I+\sqrt{E_b}\right)^2+\left(y_Q+\sqrt{E_b}\right)^2}{2\sigma_G^2}\right]}+\alpha e^{-\left[\frac{\left(y_I+\sqrt{E_b}\right)^2+\left(y_Q+\sqrt{E_b}\right)^2}{2\sigma_I^2}\right]}\right)dy_I dy_Q, \quad (3.31)$$

The conditional probability at the relay can be calculated by following the same procedure. By substituting (3.28) - (3.31) into (3.26), the theoretical error probability P_{qr}^{GMM} at the relay of PNC on impulsive noise channels with QPSK modulation can be derived as:

$$P_{qr}^{GMM} = (1-\alpha)P_{qr}^{G} + \alpha P_{qr}^{I} \tag{3.32}$$

where $P_{qr}^{G} = \frac{3}{4} erfc\left(\sqrt{\frac{E_b}{N_G}}\right)$ and $P_{qr}^{I} = \frac{3}{4} erfc\left(\sqrt{\frac{E_b}{N_I}}\right)$ are the bit error probability of AWGN and impulsive noise respectively. Thus, the error probability of transmitting signals from node to node with PNC on impulsive noise channels p_{qn}^{GMM} can be expressed as:

$$p_{qn}^{GMM} = (1-P_{GMM})P_{qr}^{GMM} + \left(1-P_{qr}^{GMM}\right)P_{GMM} - 2P_{GMM}P_{qr}^{GMM}. \tag{3.33}$$

The comparison of simulated BER to theoretical BER of uncoded PNC on the impulsive channel is shown in Fig 3.11.

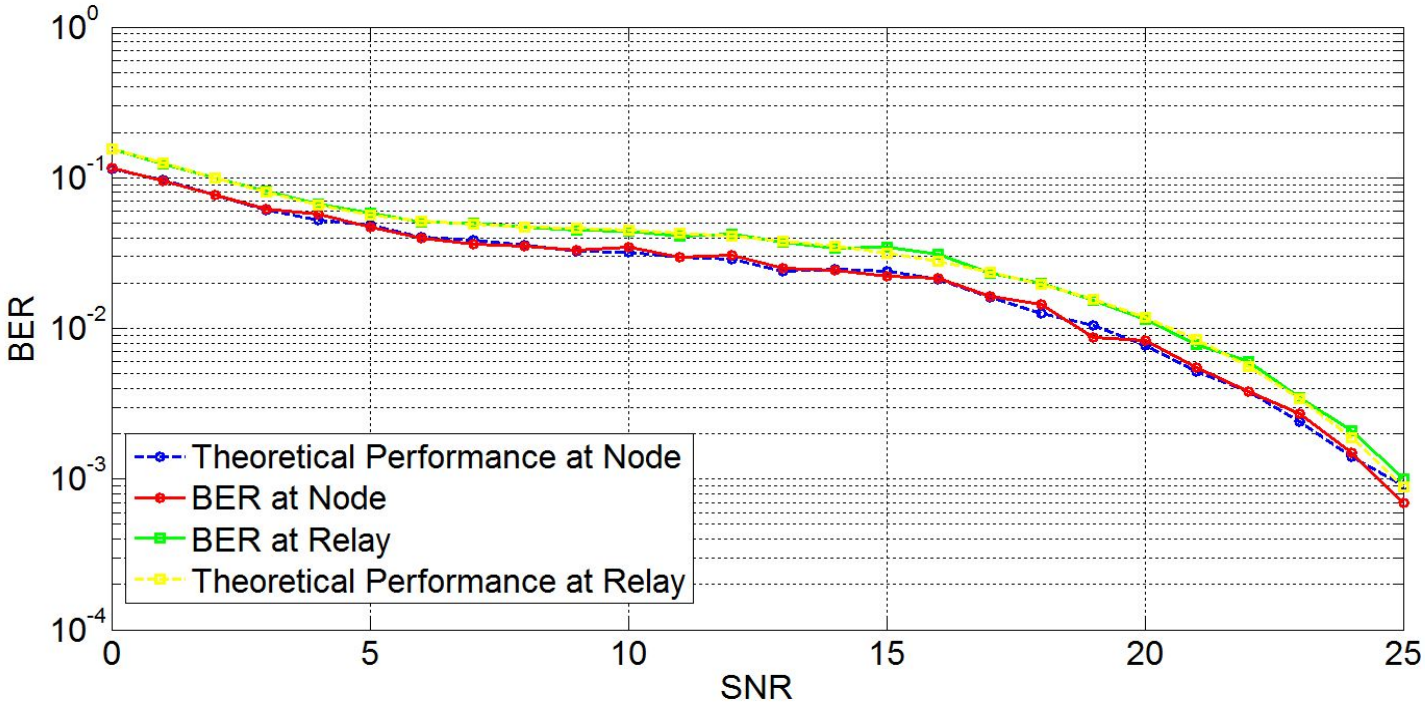


Figure 3.11 Comparison of simulated BER to theoretical BER, P_{qr}^{GMM} and p_{qn}^{GMM} at the relay/nodes on uncoded QPSK PNC over impulsive channel, $a = 0.1$.

From Fig 3.11 it can be seen that the simulation results match closely with the theoretical error probability. The performance of decoding at the relay on PNC (green curve) is slightly better than decoding at the destination node (red curve), with approximately a 1.5dB advantage. With ten percent of impulsive noise in the channel, the BERs of the system are significantly affected by the channel noise compared to the AWGN channel in Fig 3.8. The comparison verifies the expressions of the theoretical BER performance of PNC with uncoded QPSK on impulsive channels, which will support the work presented in later chapters.

3.4 Construction of Binary Convolutional Codes

Elias first proposed the class of binary convolutional codes in 1955 [65]. The major difference between convolutional codes and block codes is that block codes are only able to encode a fixed length of information bits. However, convolutional codes can encode a continuous stream of information bits. Another advantage of the convolutional codes is their simplicity: convolutional codes have a much simpler trellis than block codes. Thus, convolutional codes have been widely applied since 1955. In this section, we are going to introduce the basic principles of convolutional codes combined with PNC and evaluate the theoretical performance of convolutional codes.

3.4.1 Convolutional Encoder

There are two kinds of convolutional codes encoder that are commonly used to encode information bits: Non-Systematic convolutional (NSC) codes and Recursive Systematic Convolutional (RSC) codes [74]. NSC codes have the encoder structure shown in Fig 3.12. The generator polynomials $G = [g_1(D), g_2(D) \dots]$ specify the different types of NSC code.

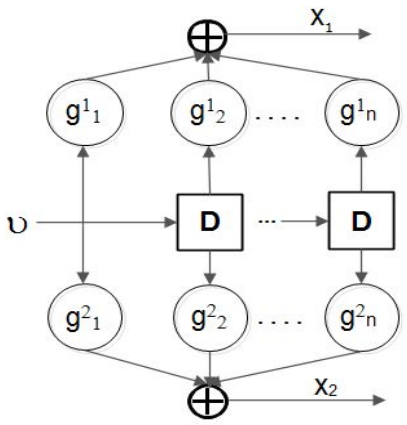


Figure 3.12 Encoder structures of NSC Codes.

As shown in Fig 3.12, the generator polynomials can be expressed as:

$$g^1(D) = [g_1^1 + g_2^1(D) + \cdots + g_n^1(D^{n-1})\,]$$
$$g^2(D) = [g_1^2 + g_2^2(D) + \cdots + g_n^2(D^{n-1})\,] \qquad (3.34)$$

where D are the memory elements, $g^1(D)$ and $g^2(D)$ used to obtain the output x_1 and x_2. Thus, for input information bits of length L, the output coded bits have a length of $\frac{1}{R_c}(L+v)$, where R_c denotes the code rate, v is the number of memory elements and the extra $R_c v$ coded bits occur due to the v bits added at the end of the information bit sequences to reset the memory elements, which is known as terminating the code.

3.4.2 Decoding of Binary Convolutional Codes

There is more than one choice of decoder for decoding convolutional codes, including majority logic decoder [75], the sequential decoder [76], the Viterbi decoder [77] and the BCJR decoder [78]. In this thesis we use MAP decoder or BCJR algorithm [79], since the MAP decoder achieves better performance at low SNR. The criterion of MAP decoding is given by:

$$\hat{x} = \arg\max_{x} P(x|y), \tag{3.35}$$

where $P(x|y)$ is the *a posteriori* probability (APP) of the transmitted symbol x given the received codeword y. Considering the $\hat{x} \in \{\pm 1\}$ due to BPSK mapping, then (3.35) can be simplified to:

$$\hat{x} = sign[L(x)], \tag{3.36}$$

where $L(\hat{x})$ is the log-likelihood ratio (LLR), which is defined as:

$$L(x|y) = \log\left[\frac{P(x=+1|y)}{P(x=-1|y)}\right]. \tag{3.37}$$

and can be extended to:

$$L(x|y) = \log\left[\frac{\sum_{\acute{s}-s\in S_1} P(x=+1|y)}{\sum_{\acute{s}-s\in S_0} P(x=-1|y)}\right]. \tag{3.38}$$

The s and $\acute{s}$ represent the current state and next state respectively in the trellis diagram, where $\acute{s} - s \in S_1$ corresponds to the input $x = +1$, and $\acute{s} - s \in S_0$ corresponds to the input $x = -1$. Thus, two pdfs $p(x=+1|y)$ and $p(x=-1|y)$ can be factored as:

$$P(x|y) = A_{k-1}(\acute{s})\gamma_k(\acute{s}, s)\, B_k(s), \tag{3.39}$$

where *k* is the index of the *k*th data, A and B are the trace forward and trace backward metrics respectively, γ is the branch metric. The trace metrics are defined as:

$$A_k(s) = \sum_{\acute{s}} \gamma_k(\acute{s}, s)\, A_{k-1}(\acute{s}), \tag{3.40}$$

$$B_{k-1}(\acute{s}) = \sum_{s} \gamma_k(\acute{s}, s)\, B_k(s). \tag{3.41}$$

The recursion of A and B is initialized based on the assumption of the encoder of convolutional code being initialized to the all-zero state and terminated to the all-zero state at $k = L$:

$$A_k(s) = \begin{cases} 1 \text{ when } s = 0 \\ 0 \text{ when } s \neq 0 \end{cases} k = 0,1 \dots L \tag{3.42}$$

$$B_k(s) = \begin{cases} 1 \text{ when } s = 0 \\ 0 \text{ when } s \neq 0 \end{cases} k = L, L-1 \dots 0 \tag{3.43}$$

where L is the length of the received information signal. The branch metric γ is defined as:

$$\gamma_K(\acute{s}, s) = \frac{P(x_k)}{2\pi\sigma^2} e^{\left(-\frac{\|y_k - c_k\|^2}{2\sigma^2}\right)} \tag{3.44}$$

where σ^2 is the noise variance. For long codeword lengths, the BCJR algorithm can be numerically unstable [80], so now the metrics are considered in the logarithmic domain. This is known as the log-BCJR or log-MAP algorithm. The way to compute metrics A, B can be expressed as:

$$A_k(s) = \log \sum_{\acute{s}} e^{\gamma_k(\acute{s},s) A_{k-1}(\acute{s})}, \tag{3.45}$$

$$B_{k-1}(\acute{s}) = \log \sum_{s} e^{\gamma_k(\acute{s},s) B_k(s)}. \tag{3.46}$$

$$\gamma_k(\acute{s}, s) = \log \frac{P(x_k)}{2\pi\sigma^2} - \left(-\frac{\|y_k - x_k\|^2}{2\sigma^2}\right) \tag{3.47}$$

As we know,

$$\log(e^x + e^y) = \max\{x, y\} + \log\left(1 + e^{-|x-y|}\right). \tag{3.48a}$$

This can be approximated to

$$\log(e^x + e^y) \approx \max\{x, y\} \tag{3.48b}$$

Thus, (3.45) to (3.47) can be rewritten as:

$$A_k(s) = \max_{\acute{s}}\{A_{k-1}(\acute{s}) + \gamma_k(\acute{s}, s)\}, \tag{3.49}$$

$$B_k(\acute{s}) = \max_{s}\{\ B_k(s) + \gamma_k(\acute{s}, s)\}, \tag{3.50}$$

$$\gamma_k(\acute{s}, s) = -\frac{\|y_k - x_k\|^2}{2\sigma^2} \tag{3.51}$$

Moreover, (3.38) can also be rewritten as:

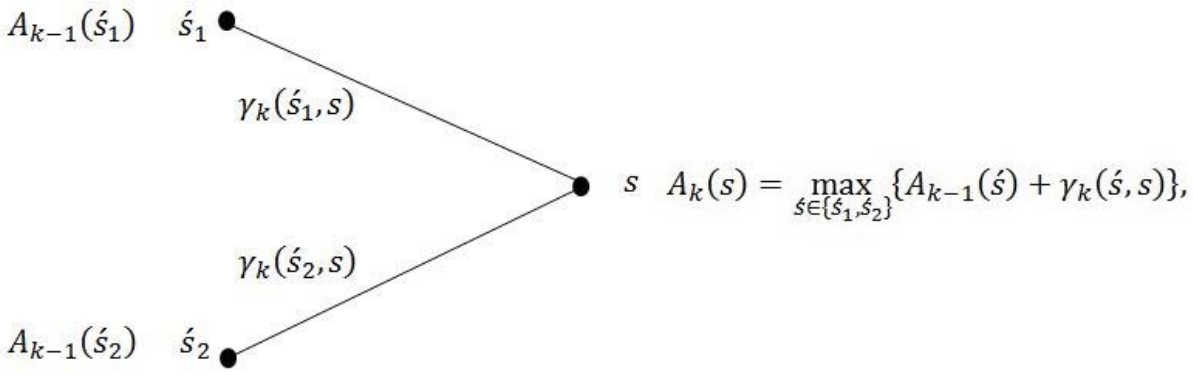
$$L(x) = \log\left[\frac{\sum_{\acute{s}-s\in x_+}\ A_{k-1}(\acute{s}) + \gamma_k(\acute{s}, s) +\ B_k(s)}{\sum_{\acute{s}-s\in S_-}\ A_{k-1}(\acute{s}) + \gamma_k(\acute{s}, s) +\ B_k(s)}\right]$$

$$= \max_{\acute{s}-s\in x_+}\ [\ A_{k-1}(\acute{s}) + \gamma_k(\acute{s}, s) +\ B_k(s)]$$

$$-\max_{\acute{s}-s\in x_-}\ [\ A_{k-1}(\acute{s}) + \gamma_k(\acute{s}, s) + B_k(s)]. \tag{3.52}$$

If the max operation defined in (3.48a) is used, then the algorithm is called the log-MAP algorithm. However, if the max operation defined in (3.48b) is used, then it is called the Max log-MAP algorithm.

In order to show the procedure of determining the trace metrics, a typcial calculation involving two branches in the trellis diagram that converge at a node is shown in Fig 3.13.



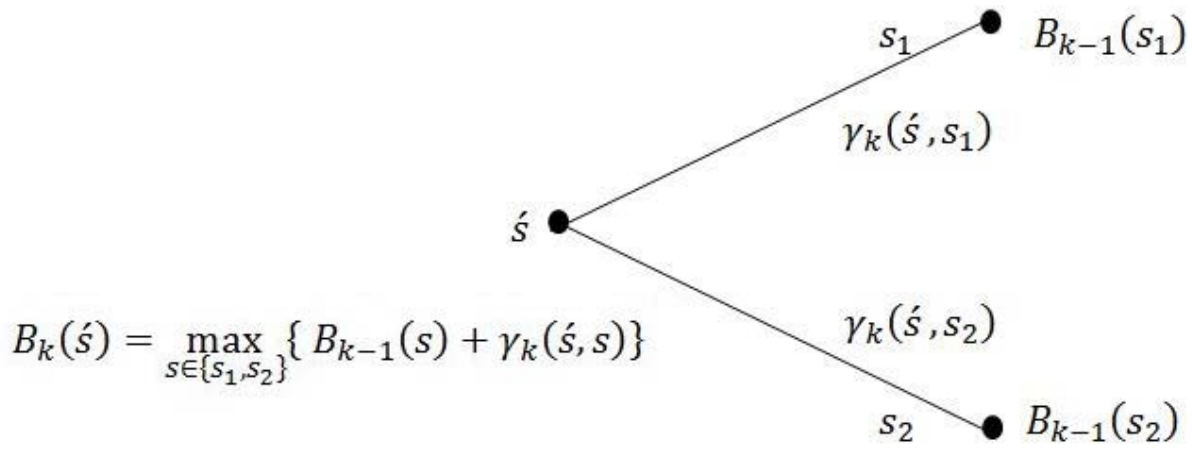


Figure 3.13 The Illustration of trace forward metric A and backward metric B of the MAX-log MAP decoder

3.4.3 Theoretical Performance of Binary Convolutional Codes

We assume without loss of generality that the all-zero sequence is the input to the encoder and the rate $\frac{1}{2}$ $(7,5)_8$ convolutional code is chosen as the example code to introduce the following analysis. The state table of the rate $\frac{1}{2}$ $(7,5)_8$ convolutional code is shown in Table 3.3. Based on the state table, it is possible to draw a signal-flow graph of the convolutional code. The Hamming distance properties and BER performance of the convolutional code can be obtained from a signal-flow graph, as shown in Fig 3.14.

Input	Initial State	Next State	Output
0	00	00	00

1	00	01	11
0	01	11	01
1	01	10	10
0	10	01	10
1	10	00	11
0	11	10	01
1	11	11	10

Table 3.3. State table of rate $\frac{1}{2}(7,5)_8$ convolutional code.

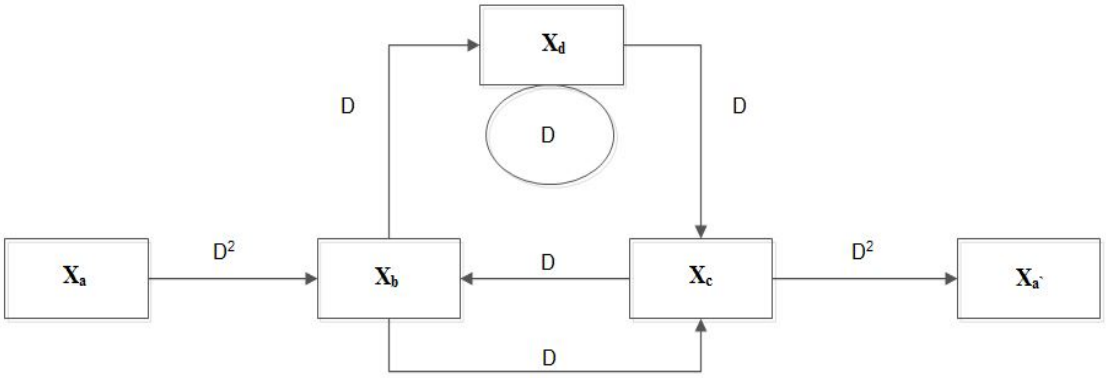


Figure 3.14 signal-flow graph of rate $\frac{1}{2}(7,5)_8$ convolutional code.

Let X_a= state 00, X_b= state 10, X_c= state 01 and X_d= state 11. Thus, based on the signal-flow graph, we can write four state equations:

$$X_b = D^2 X_a + X_c, \tag{3.53}$$

$$X_c = D\,X_d + DX_b, \tag{3.54}$$

$$X_d = D\,X_d + DX_b, \tag{3.55}$$

$$X_{a'} = D^2 X_c, \tag{3.56}$$

By definition the transfer function T(D) is defined as:

$$\mathrm{T(D)} = \frac{X_{a'}}{X_a}. \tag{3.57}$$

By submitting (3.53)-(3.56) to Eqn (3.57), we have:

$$\mathrm{T(D)} = \frac{D^5}{1 - 2D}. \tag{3.58}$$

The exponent of *D* shows the distance of the sequence of encoded bits for that path from the all-zero sequence. Then the transfer function T(D) can be evaluated by performing the long-division:

$$\mathrm{T(D)} = D^5 + 2D^5 + \cdots \tag{3.59}$$

The bound on the bit-error probability of the rate $\frac{1}{2}(7,5)_8$ convolutional code P_{bc} can be expressed as [81]:

$$P_{bc} \leq \frac{1}{K} \sum_{w} \sum_{d_{free}}^{N} T_d P_{channel}. \tag{3.60}$$

where $P_{channel} = \frac{1}{2} erfc\left(\sqrt{\frac{E_b}{N_0}}\right)$, *N* is the codeword length and *K* is the information bits length, *R* is the code rate. Thus by substituting (3.59) into (3.60), the bound on the bit error probability of a rate $\frac{1}{2}(7,5)_8$ convolutional code can be calculated. A comparison of simulated BER to theoretical bound of rate $\frac{1}{2}(7,5)_8$ convolutional code is shown in Figure 3.15. From the figure it can be seen that the bound matches the simulated BER closely, and proves the derivation is correct.

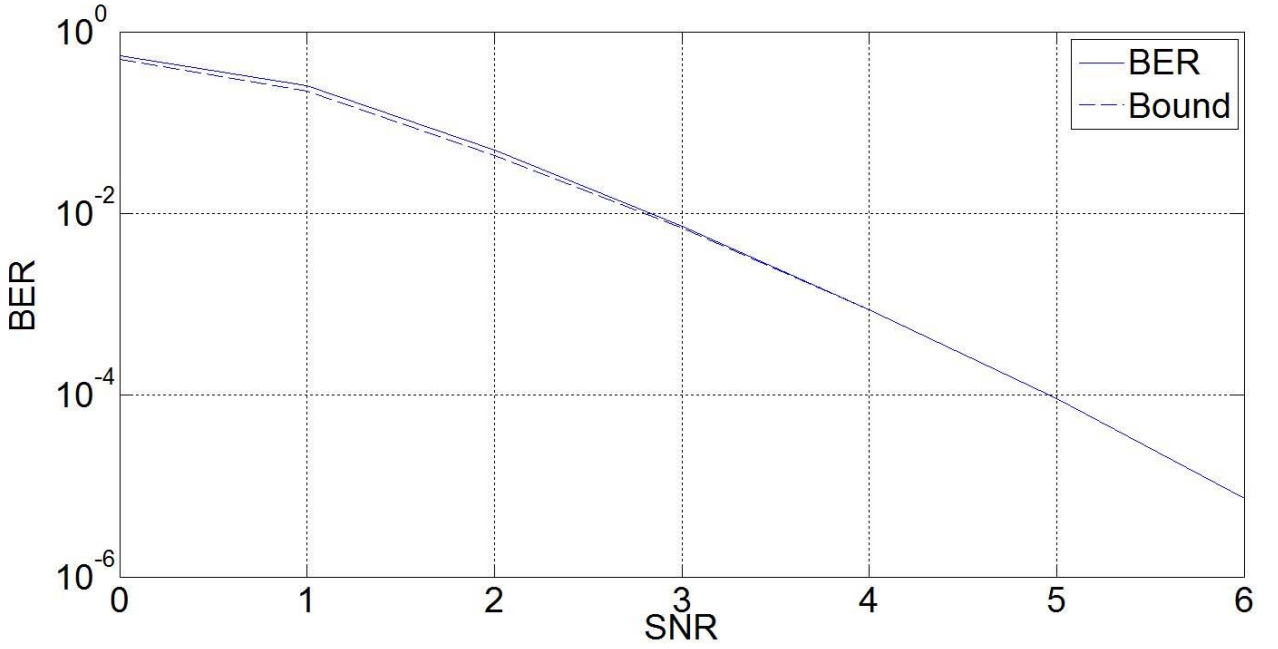


Figure3.15 Comparison of simulated BER to theoretical bound of rate $\frac{1}{2}(7,5)_8$ convolutional code

3.5 Finite Fields

The nonbinary coding schemes used in chapters 5 and 6 in this thesis have elements defined in finite fields, denoted as GF(q). A finite field is a field containing a finite number of elements and the number of elements in the finite field is called the *order* [75]. The elements in the field can be added, subtracted, multiplied or divided. A finite field of order q exists if the order q is a prime power p^{α}, where p is a prime number and α is a positive integer [76]. A finite field in which the element can takes q different values is referred to as GF(q). In the work of this thesis, we are focused on the field of four and field of sixteen, that is GF(4) and GF(16). Table 3.4 and Table 3.5 list the mapping from binary bits to nonbinary symbols. It needs to be denoting here, the primitive polynomial is $\beta^2+\beta+1=0$ gives $\beta^2=\beta+1$ by modulo-2 addition in a field of $F=$

$\{0,1, \beta, \beta^2, \ldots, \beta^j, \ldots\}$. Let $p(X)$ be the primitive polynomial of degree z over GF(2), and assume $p(\beta) = 0$. Since $p(X)$ divides $X^{2^z-1} + 1$, we have $X^{2^z-1} + 1 = q(X)p(X)$ where $q(X)$ can be regarded as a polynomial of β over GF(2). Thus, if $X = \beta$, we have $\beta^{2^z-1} + 1 = q(\beta)p(\beta)$. Since $p(\beta) = 0$, we have $\beta^{2^z-1} + 1 = 0 \Longrightarrow \beta^{2^z-1} = 1$. Therefore, the field can be expressed as $F = \{0,1, \beta, \beta^2, \ldots, \beta^{2^z-2}\}$, that is a Galois field of 2^z elements GF(2^z).

Binary	GF(4)	Polynomial
00	0	0
01	1	1
10	β	β
11	β^2	$\beta + 1$

Table 3.4. Mapping of GF(4), primitive polynomial is $\beta^2 + \beta + 1 = 0$.

Binary	GF(16)	Polynomial
0000	0	0
0001	1	1
0010	β	β
0100	β^2	β^2
1000	β^3	β^3
1001	β^4	$\beta^3 + 1$
1011	β^5	$\beta^3 + \beta + 1$
1111	β^6	$\beta^3 + \beta^2 + \beta + 1$

0111	β^7	$\beta^2 + \beta + 1$
1110	β^8	$\beta^3 + \beta^2 + \beta$
0101	β^9	$\beta^2 + 1$
1010	β^{10}	$\beta^3 + \beta$
1101	β^{11}	$\beta^3 + \beta^2 + 1$
0011	β^{12}	$\beta + 1$
0110	β^{13}	$\beta^2 + \beta$
1100	β^{14}	$\beta^3 + \beta^2$

Table 3.5. Mapping of GF(16), primitive polynomial is $\beta^4 + \beta^3 + 1 = 0$.

There are two operations in GF(q) commonly used in the encoding and decoding of non-binary codes: addition and the multiplication. The addition and multiplication tables of GF(4) are shown in table 3.6 and 3.7 respectively.

+	**0**	**1**	**β**	**β^2**
0	0	1	β	β^2
1	1	0	β^2	β
β	β	β^2	0	1
β^2	β^2	β	1	0

Table 3.6. Addition table for GF(4).

×	**0**	**1**	**β**	**β^2**
0	0	0	0	0
1	0	1	β	β^2

β	0	β	β^2	1
β^2	0	β^2	1	β

Table 3.7. Multiplication table for GF(4).

3.6 Conclusions

In this chapter, an introduction to TWRC employing PNC at the relay is given, including a comparison of the simulated BER performance with the theoretical BER performance of uncoded PNC at the relay and destination nodes. Furthermore, the basic concepts of impulsive noise channels based on GMM are presented with the noise distribution expression and noise variance analysis. A brief introduction of encoding and decoding convolutional codes is provided with a comparison of simulated BER to theoretical bound. Finally, the fundamental knowledge of finite fields is presented explaining how to add and multiply finite field elements in order to have a better understanding of encoding and decoding nonbinary channel codes in future chapters

Chapter 4

PNC Combined with Iterative Trellis Decoders

4. Binary Turbo codes and Trellis BICM-ID

4.1 Introduction

Shannon's channel coding theory states that as the block size of a random code increases, it will approach its optimal performance, known as the Shannon limit. This initiated a new research area with the aim of constructing codes that could achieve the Shannon limit but with a reasonable complexity. In 1993, turbo codes, or parallel concatenated convolutional codes (PCCCs), were introduced by Berrou et.al and Glavieux later [58]. It can be said that turbo codes are one of the most significant breakthroughs in error control coding, achieving near-Shannon limit performance with a decoder complexity that enables it to be implemented in hardware. One of the main factors for this excellent performance is the iterative turbo decoder, which comprises two component soft-input-soft-output decoders in series passing prior information to each other to achieve further improvements in performance after each iteration. The turbo principle has also been applied to other coding schemes, in particular trellis bit-interleaved coded modulation with iterative decoding (BICM-ID), which comprises a single convolutional encoder and interleaver, but decoding is achieved iteratively by passing prior information between a demapper and SISO decoder.

In this chapter, PNC is combined with turbo codes and the performance is evaluated when the channels between source/destination nodes and relay node are impulsive. The additive impulsive noise channels are modelled using the well-known Gaussian mixture model (GMM), as described in chapter 3. The turbo encoder and decoder are first explained and this is followed by an explanation of the combination of PNC with turbo codes. A detailed analysis of the performance of PNC with turbo codes on impulsive noise channels is then presented. The extrinsic information (ExIT) chart showing the behaviour of the iterative turbo decoder at the relay of a TWRC with impulsive noise is derived. Furthermore, a upper bound on the BER performance of turbo codes is derived to accurately determine the error floor for different impulsive noise channels.

Next, an investigation into PNC combined with trellis BICM-ID on additive impulsive noise channels is presented. The encoder and iterative demapper/decoder scheme is explained and how PNC is combined with trellis BICM-ID. The ExIT charts of the iterative demapper/decoder scheme for trellis BICM-ID are derived at the relay of a

TWRC employing PNC with additive impulsive noise channel. The chapter is concluded by comparing PNC combined with turbo codes and trellis BICM-ID in terms of performance and complexity.

4.1.1 Turbo Code Encoder

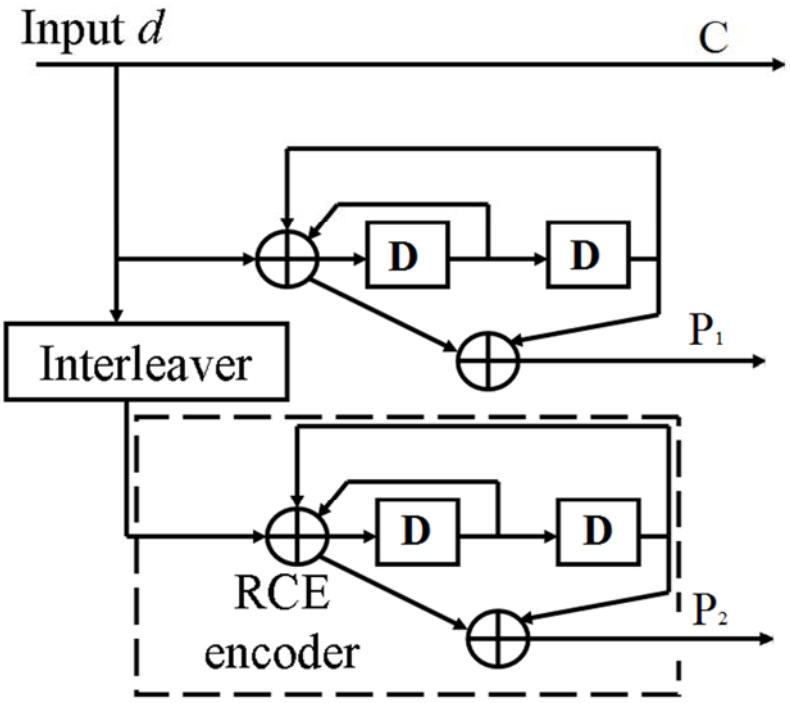


Figure 4.1 Encoder structure of turbo code.

The turbo code encoder is a parallel concatenation of two identical recursive systematic convolutional (RSC) codes encoders separated by an interleaver, as shown in Figure 4.1. The interleaver is usually a pseudorandom interleaver that interleaves the message for encoder 2. The combination of two RSC encoders with an interleaver produces a codeword with a high hamming weight, and the interleaver makes different codewords relatively sparse, named *multiplicity*, increase the coding gain of turbo code.

The standard turbo code encoder polynomials can be expressed as:

$$G(D) = \left[1, \quad \frac{g_{(2)}(D)}{g_{(1)}(D)}\right] \tag{4.1}$$

where $g_{(1)}(D)$ and $g_{(2)}(D)$ are the feedforward and feedback polynomials respectively, e.g. $\left(1, \frac{5}{7}\right)_8$ RSC code can be represented by the binary strings $g_{(1)} = 101$ and $g_{(2)} = 111$ for which $g_{(1)}(D) = 1 + D + D^2$ and $g_{(2)}(D) = 1 + D^2$. It can be seen that the general codeword produced by turbo encoder has a rate of 1/3, which can be increased to a higher rate of 1/2 with a puncture matrix of $P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

4.1.2 Puncturing

It can be seen that the codeword generated by the turbo encoder has a rate of 1/3, but this can be increased to higher code rates by removing certain patterns of bits from the codeword to decrease the block size. This is called puncturing and for turbo codes it is only applied to the parity-check bits from each RSC encoder. A puncture matrix **P** is used to determine which bits are removed. For example, the puncture matrix $\mathbf{P} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ increases the code rate of a rate 1/3 turbo code to a rate 1/2 turbo code by removing the even indexed bits from the first RSC encoder and the odd indexed bits from the second RSC encoder. The puncture matrix must also be known for the decoder process so that it knows which bits in the codeword have been removed.

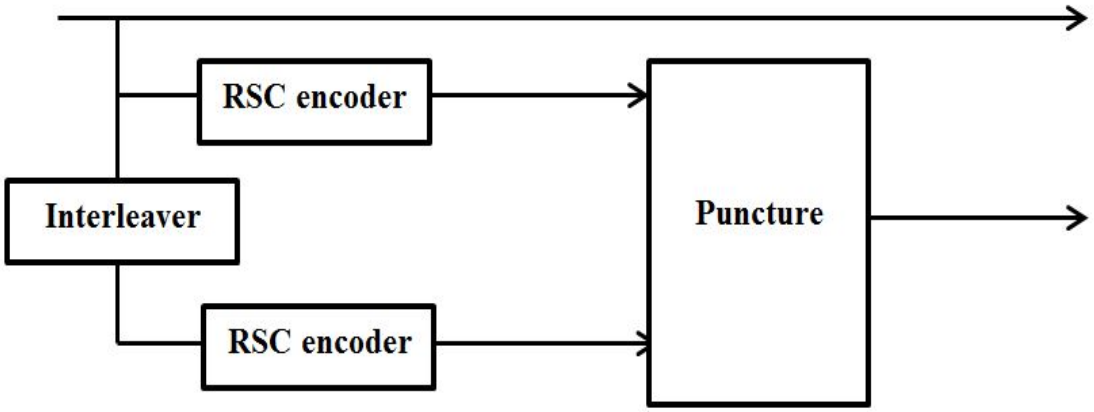


Figure 4.2 Encoder structure of turbo code with a puncture.

4.1.3 Turbo Decoder

The turbo code decoder in this thesis is constructed from two soft-in-soft-out (SISO) Max log-MAP decoders that exchanges prior information between each decoder in every decoding iteration loop to converge to an optimal performance, as shown in Fig. 4.3. In a decoding iteration, the first decoder takes in LLR values of the received message and parity-check bits and LLR values from the second decoder. The first decoder updates the LLRs of the message symbols and then extrinsic LLRs are extracted by subtracting the original LLRs of the message symbols and the a priori LLRs from the second decoder. Then the second decoder takes in the original interleaved LLRs of the message symbols and the interleaved extrinsic LLRs from the first decoder, which becomes the a priori LLRs. The second decoder updates the interleaved LLRs of the message symbols and extrinsic LLRs are extracted by subtracting the original interleaved LLRs of the message symbols and the a priori LLRs from the first decoder. Finally, the extrinsic LLRs are deinterleaved and become the new a priori LLRs for the first decoder. This completes one decoding iteration. After a certain number of iterations, the turbo decoder performance either converges until no more errors are present or fails if too many errors are present.

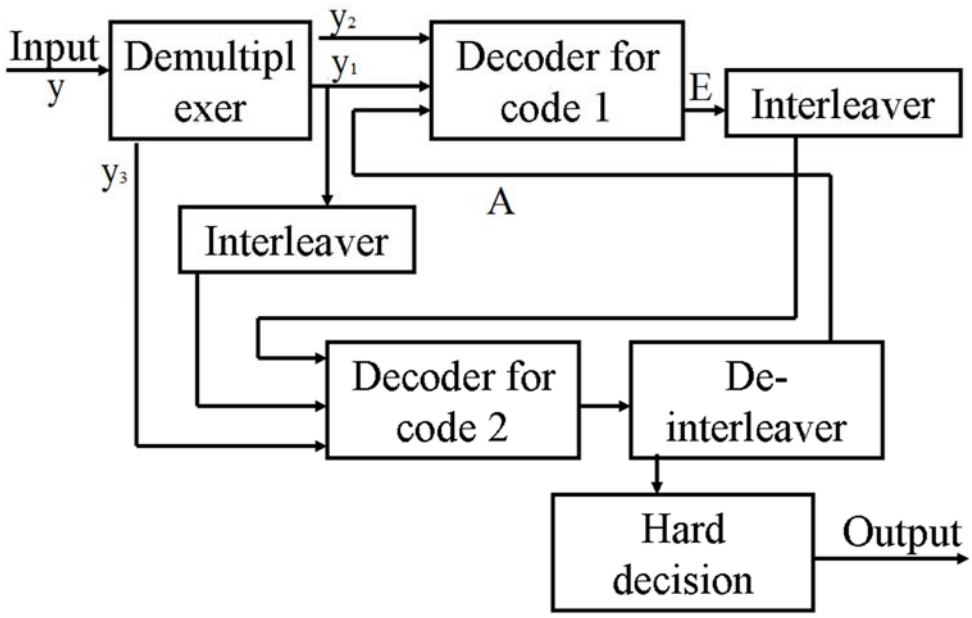


Figure 4.3 The turbo decoder

To decode the soft received symbols using the MAX-log-MAP algorithm, three metrics A, B and γ are defined. The A and B metrics are the trace-forward and trace-backward metrics and γ is the branch metric. A and B are initialized as in (3.42)-(3.43), but γ is now defined as:

$$\gamma_K(\acute{s}, s) = \frac{x_K L_e(x_K)}{2} - \frac{\|y_K - c_K\|^2}{2\sigma^2}. \tag{4.2}$$

where $L_e(x_K)$ is the extrisic information of x_K, K is the index, s is the present state and $\acute{s}$ is the next state and σ^2 is the noise variance.

Thus, the output LLR is calculated as:

$$L(x) = \max_{\acute{s}-s \in S_1} \{ A_{K-1}(\acute{s}) + \gamma_K(\acute{s}, s) + B_K(s)\}$$
$$- \max_{\acute{s}-s \in S_0} \{ A_{K-1}(\acute{s}) + \gamma_K(\acute{s}, s) + B_K(s)\} . \tag{4.3}$$

where S_+ is the set of all state transitions corresponding to x = +1 and S_- is the set of all state transitions corrresponding to x = -1. By substituting (3.42)-(3.43) and (4.2) into (4.3), we have:

$$L(x|y) = L_e(x_K) + \max_{\acute{s}-s \in x_+} \left\{ A_{K-1}(\acute{s}) + \frac{y_K c_K}{\sigma^2} + B_K(s) \right\}$$
$$- \max_{\acute{s}-s \in x_-} \left\{ A_{K-1}(\acute{s}) + \frac{y_K c_K}{\sigma^2} + B_K(s) \right\} + \frac{2y_K}{\sigma^2} . \tag{4.4}$$

In (4.4), the first term is the extrinsic LLR received from the other decoder, the second and third terms are the *a priori* information send to the other decoder, and the fourth term is directly received from the noise channel, known as the channel reliability. The performance of a turbo code with $\left(1, \frac{5}{7}\right)_8$ RSC encoders on the AWGN channel is shown in Fig. 4.4, for one iteration up to 10 iterations. Observe how the gain in performance becomes smaller with each iteration until it converges. Also notice that an error floor appears at around a BER of 10^{-5}, which is a characteristic of turbo codes.

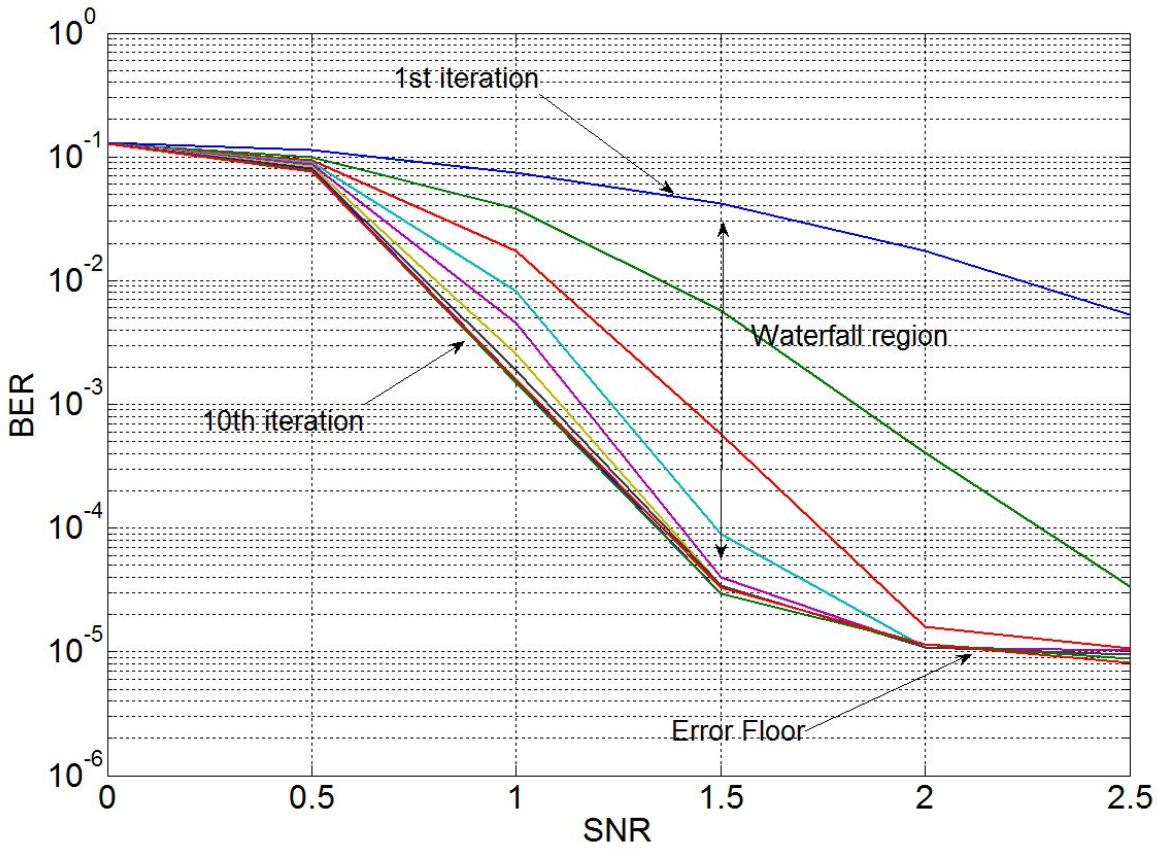


Figure 4.4 BER performance of $(7,5)_8$ Turbo code on AWGN channel.

4.2 PNC combined with Turbo Codes

Based on the review carried out in chapter 2, it appears that the effect of impulsive noise on a conventional TWRC employing PNC has not been considered. Of particular interest is the effect of impulsive noise on the iterative decoder employed at the relay. When evaluating the performance of an iterative decoding scheme, it is important to investigate the convergence behaviour of the iterative decoding algorithm, which can be achieved using ExIT charts [9]. When there are two decoders exchanging information, the behaviour can be plotted with respect to each decoder. The process of exchanging information is represented in a chart that depicts the transfer of mutual information between the *a priori* information and the extrinsic information passed between these decoders. To obtain an ExIT chart for the turbo code affected by impulsive noise, we need to know the probability density function (pdf) of the noise. To achieve this, the Gaussian mixture model (GMM) has been selected [10][11], which was defined in chapter 3.

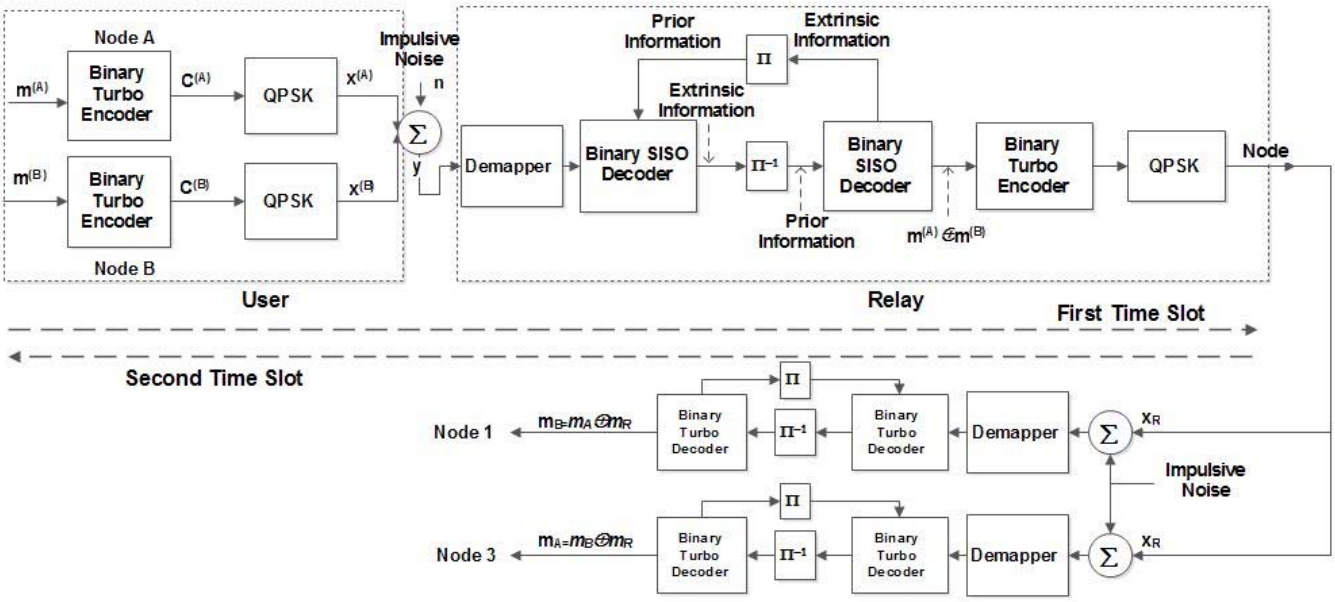


Figure 4.5 System model of turbo coded PNC over impulsive noise channel, where the channel encoder is the required turbo encoder, e.g. (7,5)$_8$.

The performances of rate $\frac{1}{2}$ and rate $\frac{1}{3}$ turbo codes are evaluated on the conventional TWRC employing PNC, where the uplink and downlink channels are additive impulsive noise channels. Furthermore, ExIT charts are presented to analyse the effect of impulsive noise on the iterative decoding algorithms at the relay and to validate the simulations results in following sections.

The system model of turbo coded PNC on impulsive noise channels is shown in Figure 4.5. Let $m_A \in \{0,1\}^k$ and $m_B \in \{0,1\}^k$ be the k-bit binary messages sent from node A and B. The information sequences are encoded resulting in $c_A \in \{0,1\}^n$ and $c_B \in \{0,1\}^n$, where n is the block size of the codes.

The received information sequence at the relay can be expressed as:

$$y = x_A + x_B + \eta, \tag{4.9}$$

where x_A and x_B are complex QPSK symbols transmitted from nodes A and B respectively, η is the noise added at the relay and $(x_A, x_B) \in \{1+\mathrm{j}, 1-\mathrm{j}, -1-\mathrm{j}, -1+\mathrm{j}\}$. The sum of the two transmitted QSPK signals x_R can have nine possible complex values, as shown in Figure 4.6.

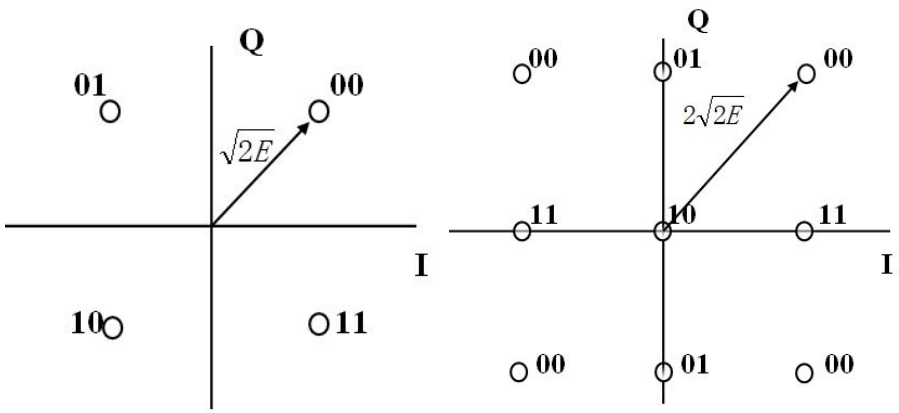


Figure 4.6. Constellation diagram of anti-Gray mapped QPSK.

The relay must then determine the log-likelihood ratio (LLR) of y given that $x_R = x_A + x_B$ was transmitted, $L(y|x_A + x_B)$. This is decoded at the relay to give the message $m_R = m_A \oplus m_B$, where $\oplus$ is the XOR operation. The decoded message is then re-encoded to give $c_R = c_A \oplus c_B$, which is mapped to a QPSK constellation and broadcast back to nodes *A* and *B*. At nodes *A* and *B*, the received signal is decoded to obtain m_R, where node *A* can obtain m_B by performing the XOR of m_R with its known binary message m_A and vice versa. To show that the codeword $c_1 + c_2$ can be obtained by encoding the message $m_1 + m_2$, consider a general encoder with generator matrix **G**. We therefore have:

$$c_1 = m_1 \cdot \mathrm{G} \tag{4.5}$$

$$c_2 = m_2 \cdot \mathrm{G} \tag{4.6}$$

So, the sum $c_1 + c_2$ is:

$$c_1 + c_2 = m_1 \cdot \mathrm{G} + m_2 \cdot \mathrm{G} \tag{4.7}$$

which can be simplified to:

$$c_1 + c_2 = (m_1 + m_2) \cdot \mathrm{G} \tag{4.8}$$

In other words, the message $m_1 + m_2$ can be recovered from the codeword $c_1 + c_2$.

The turbo decoder will decode a vector of LLR values that give a measure of the reliability of the combination of both source nodes' codewords. We choose to use the Max-log-MAP decoding algorithm for the component decoders, so at the output of each decoder, the a posteriori LLR $L(x_R|y)$ can be expressed as:

$$L(x_R|y) = L_c y \ + L_a(x_R) + L_e(x_R), \tag{4.10}$$

where

$$L_c = \frac{4\sqrt{E_B}}{N_0} \tag{4.11}$$

is the channel reliability that represents the relationship between channel outputs corresponding to the systematic bits.

$$L_a(x_R^1) = \ln\left(\frac{P(c_R = 11) + P(c_R = 10)}{P(c_R = 01) + P(c_R = 00)}\right)$$

$$L_a(x_R^2) = \ln\left(\frac{P(c_R = 11) + P(c_R = 01)}{P(c_R = 10) + P(c_R = 00)}\right) \tag{4.12}$$

where x_R^1 and x_R^2 are the first and the second bit of x_R, $L_a(x_R)$ is the a priori LLR and $L_e(x_R)$ is the extrinsic LLR of each decoder.

From the resulting nine-point constellation diagram at the relay, each received symbol y is demapped to a pair of LLR values, $L(y|m_R^1)$ and $L(y|m_R^2)$, which are a measure of the reliability of the two XORed transmitted bits, m_R^1 and m_R^2, where the superscript denotes the first or second bit and $m_R^1 = m_A^1 \oplus m_B^1$ and $m_R^2 = m_A^2 \oplus m_B^2$. Hence the LLR of y conditioned on m_R^1 can be written as [67]:

$$L(y|m_R^1) = \ln\left(\frac{P(y|m_R^1 = 1)}{P(y|m_R^1 = 0)}\right), \tag{4.13}$$

where m_R^1 is the first bit of m_R. By following the constellation diagram in Figure 4.6, the conditional probability of y given that $m_R^1 = 0$ is determined as:

$$P(y|m_R^1 = 0) = P(y|m_R = 00) + P(y|m_R = 01)$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(\sqrt{\frac{1}{2\pi\sigma^2}}\left(e^{-\frac{(y_Q-2E)^2}{2\sigma^2}} + e^{-\frac{(y_Q+2E)^2}{2\sigma^2}}\right)\left(e^{-\frac{(y_I-2E)^2}{2\sigma^2}} + e^{-\frac{(y_I+2E)^2}{2\sigma^2}} + 2e^{-\frac{(y_I)^2}{2\sigma^2}}\right)\right)dy_I dy_Q \tag{4.14}$$

where σ^2 is the noise variance, y_I is the real part of y and y_Q is the imaginary part. Similarly, the conditional probability of y given that the $m_R^1 = 1$ is determined as:

$$P(y|m_R^1 = 1) = P(y|m_R = 10) + P(y|m_R = 11)$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} 2\sqrt{\frac{1}{2\pi\sigma^2}}e^{-\frac{(y_Q)^2}{2\sigma^2}}\left(e^{-\frac{(y_I-2E)^2}{2\sigma^2}} + e^{-\frac{(y_I+2E)^2}{2\sigma^2}} + 2e^{-\frac{(y_I)^2}{2\sigma^2}}\right)dy_I dy_Q \quad (4.15)$$

Therefore, by submitting (4.14) and (4.15) into (4.13) the reliability of y can be rewritten as:

$$L(y|m_R^1) = \ln\left(\frac{P(y|m_R = 10) + P(y|m_R = 11)}{P(y|m_R = 00) + P(y|m_R = 01)}\right) \quad (4.16)$$

This can be simplified to:

$$L(y|m_R^1) = \ln\left(\cosh\frac{2Ey_Q}{\sigma^2}\right) - \frac{1}{\sigma^2} \quad (4.17)$$

The derivation can be viewed in the appendix.

Similarly, the second bit $L(y|m_R^2)$ can also be determined as:

$$L(y|m_R^2) = \ln\left(\frac{P(y|m_R = 01) + P(y|m_R = 11)}{P(y|m_R = 10) + P(y|m_R = 00)}\right) = \ln\left(\cosh\frac{2Ey_I}{\sigma^2}\right) - \frac{1}{\sigma^2} \quad (4.18)$$

Hence, based on (4.17) to (4.18), the pdf of the GMM at the relay can be derived. For example, the conditional pdfs of y given that $m_R^1 = 1$ when impulsive noise is added at the relay is:

$P_{GMM}(y|m_R^1 = 1)$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left((1-\alpha)2\sqrt{\frac{1}{2\pi\sigma_G^2}}e^{-\frac{(y_Q)^2}{2\sigma_G^2}}\left(e^{-\frac{(y_I-2E)^2}{2\sigma_G^2}}+e^{-\frac{(y_I+2E)^2}{2\sigma_G^2}}+2e^{-\frac{(y_I)^2}{2\sigma_G^2}}\right)\right.$$

$$\left.+\alpha\sqrt{\frac{1}{2\pi\sigma_I^2}}e^{-\frac{(y_Q)^2}{2\sigma_I^2}}\left(e^{-\frac{(y_I-2E)^2}{2\sigma_I^2}}+e^{-\frac{(y_I+2E)^2}{2\sigma_I^2}}+2e^{-\frac{(y_I)^2}{2\sigma_I^2}}\right)\right)dy_I dy_Q \qquad (4.19)$$

and

$P(y|m_R^1 = 0)$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(1-\alpha)\sqrt{\frac{1}{2\pi\sigma_G^2}}\left(e^{-\frac{(y_Q-2E)^2}{2\sigma_G^2}}+e^{-\frac{(y_Q+2E)^2}{2\sigma_G^2}}\right)\left(e^{-\frac{(y_I-2E)^2}{2\sigma_G^2}}+e^{-\frac{(y_I+2E)^2}{2\sigma_G^2}}+2e^{-\frac{(y_I)^2}{2\sigma_G^2}}\right)$$

$$+\alpha\sqrt{\frac{1}{2\pi\sigma_I^2}}\left(e^{-\frac{(y_Q-2E)^2}{2\sigma_I^2}}+e^{-\frac{(y_Q+2E)^2}{2\sigma_I^2}}\right)\left(e^{-\frac{(y_I-2E)^2}{2\sigma_I^2}}+e^{-\frac{(y_I+2E)^2}{2\sigma_I^2}}+2e^{-\frac{(y_I)^2}{2\sigma_I^2}}\right)dy_I dy_Q \qquad (4.20)$$

where $\sigma_G^2$and σ_I^2 are the variances of Gaussian noise variance and impulsiveness noise variance respectively. Hence, by substituting (4.19) and (4.20) into (4.16), $L_{GMM}(y|m_R^1)$ and $L_{GMM}(y|m_R^2)$ can be expressed as:

$$L_{GMM}(y|m_R^1) = \ln\left(\frac{P_{GMM}(y|m_R^1 = 1)}{P_{GMM}(y|m_R^1 = 0)}\right) \qquad (4.21)$$

$$L_{GMM}(y|m_R^2) = \ln\left(\frac{P_{GMM}(y|m_R^2 = 1)}{P_{GMM}(y|m_R^2 = 0)}\right) \qquad (4.22)$$

The extension of $L_{GMM}(y|m_R^1)$ and $L_{GMM}(y|m_R^2)$ is lengthy and can be found in the appendix.

4.3 ExIT Chart of Turbo Codes

Each component decoder in the turbo decoder receives three sets of LLRs, which are the channel reliability for the systematic bits $L_{ch}^{\tilde{x}}$, the *a priori* LLR L_a and the channel reliability for parity check bits L_{ch}^{p}. The extrinsic LLR, L_e, can be expressed as:

$$L_e = L^{total} - L_{ch}^{\tilde{x}} - L_a, \tag{4.23}$$

where the *a priori* information L_a of one decoder is the extrinsic information L_e sent from the other decoder. Therefore, we have the mutual information between the *a priori* information and the systematic message $\tilde{x}$, represented as $I(X;A)$, and the mutual information between the extrinsic information and the systematic message $\tilde{x}$, which is $I(X;E)$. The relationship between the mutual information $I(X;E)$ and the systematic message $\tilde{x}$ in AWGN channel with binary input can be expressed as:

$$I(X;E) = T\left(I(X;A), \frac{E_b}{N_0}\right) \tag{4.24}$$

The pdf of the conditional extrinsic information on systematic input X, $p_G(y|X)$, can be expressed as:

$$p_G\,(y|\tilde{x}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-\tilde{x})^2}{2\sigma^2}} \tag{4.25}$$

In order to analyse the behaviour of the turbo decoding algorithm at the relay, the ExIT chart is introduced, which tells us the number of iterations required for a decoder to converge at a particular SNR, or conversely that the system will not converge at a particular SNR. To generate the ExIT chart characteristics, it is necessary to introduce the concepts of the turbo decoder first. In the iterative decoding process, A is obtained from the other decoder. The extrinsic LLR E is produced by the turbo decoders based on the received sequence and A. According to Bayes' rule, the distribution of X at the relay can be expressed as:

$$p(y) = \sum_{\tilde{x}} p(y|X=\tilde{x})P(X=\tilde{x}) \tag{4.26}$$

where the value of $\tilde{x}$ is related to the modulation scheme, and $P(X = \tilde{x})$ is the probability. By substituting $p(y)$ into the GMM $p_G(y)$ can be obtained. By using the Kullback-Leibler distance, the mutual information between *X* and *A* can be computed as:

$$I(X;A) = D(p_{XA}||p_X p_A) \int p_{XA}(\tilde{x},y) \log_2 \frac{p_{XA}(\tilde{x},y)}{p_X(\tilde{x})p_A(y)} d\tilde{x}dy, \tag{4.27}$$

where p_{XA} is the joint probability distribution of *X* and *A*. By applying Bayes' rule we can rewrite *I(X;A)* as:

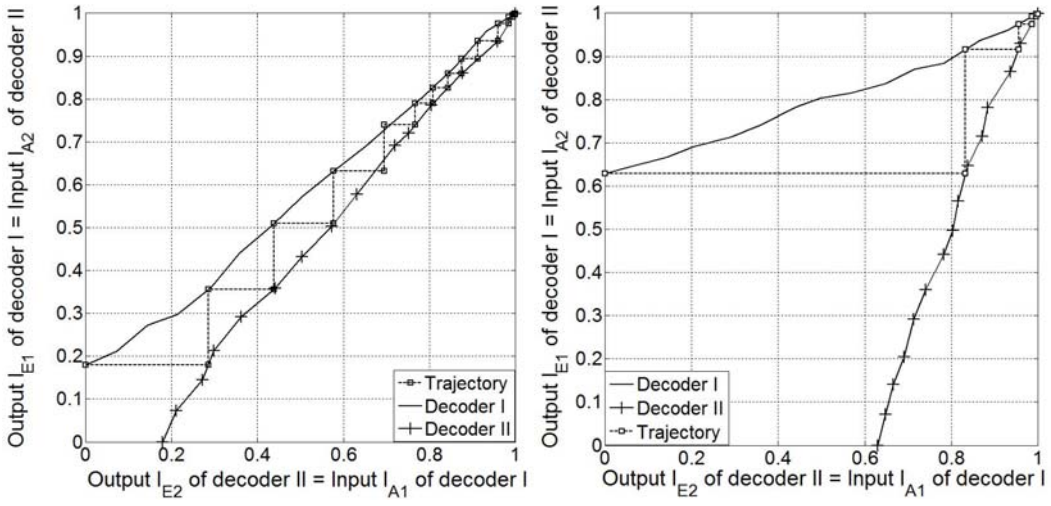
$$\begin{aligned} I(X;A) &= \int p_{A|X}(y|\tilde{x})p_X(\tilde{x}) \log_2 \frac{p_{A|X}(y|\tilde{x})p_X(\tilde{x})}{p_X(\tilde{x})p_A(y)} d\tilde{x}dy \\ &= \frac{1}{N}\int_{-\infty}^{+\infty} p_{A|X}(y|\tilde{x}) \log_2 \frac{N_s p_A(y|\tilde{x})}{\sum_{\tilde{x}\in\{S\}} p_A(y|\tilde{x})} dy. \end{aligned} \tag{4.28}$$

where N_s is the number of states of $\tilde{x}$ and $S \in \{x_R\}$. The extrinsic LLR *E* from the output of the decoder can be used to determine *I(X;E)* by generating a histogram of the extrinsic outputs. Since the distribution of *E* is not Gaussian we can compute *I(X;E)* as:

$$I(X;A) = \frac{1}{N}\int_{-\infty}^{+\infty} p_{E|X}(y|\tilde{x}) \log_2 \frac{N_s p_E(y|\tilde{x})}{\sum_{\tilde{x}\in\{S\}} p_E(y|\tilde{x})} dy. \tag{4.29}$$

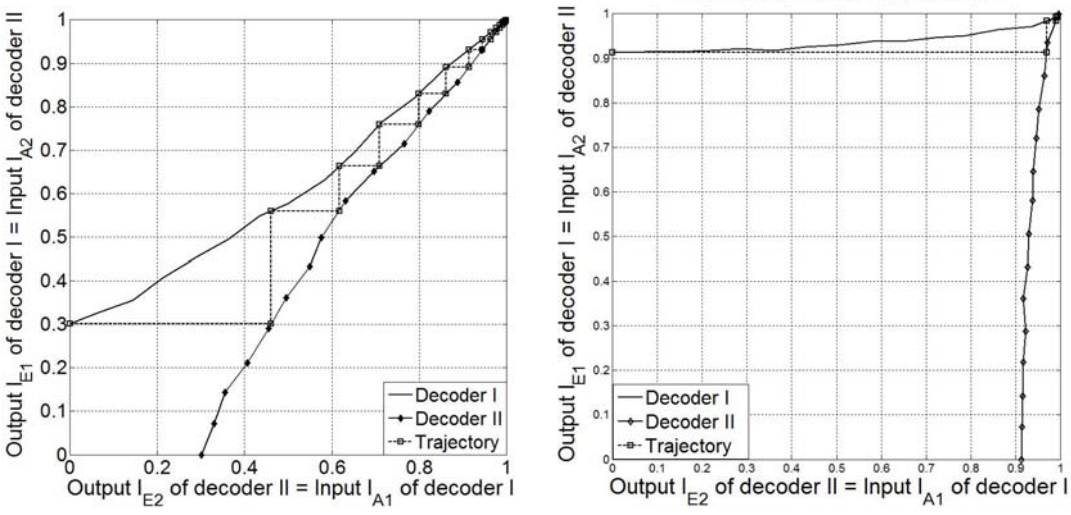
When performing the ExIT chart analysis, we chose values of α from 0.01 to 0.1 to analyse the convergence behaviour of the turbo codes at the relay. From Figure. 4.7(a) it can be seen that the lowest SNR, or pinch-off SNR, where the rate $\frac{1}{2}$ turbo coded PNC system converges is 3.1dB. The trajectory in the ExIT chart indicates that approximately 12 or 13 iterations are required to achieve convergence. Similarly, Figure.4.8 (a) shows the ExIT charts for the rate $\frac{1}{3}$ turbo code and we observe that the pinch-off SNR limit for the turbo code is 2.7dB with approximately ten iterations required. Figure.4.7 (b) shows the ExIT charts of the rate $\frac{1}{2}$ turbo code when α=0.1. In this case, the pinch-off SNR limit of the turbo code is 16.5dB, and it takes about four iterations for the decoder to achieve convergence. Again, we also see this behaviour in the ExIT charts of the rate $\frac{1}{3}$ turbo

codes are shown in Figure.4.8 (b), where the rate $\frac{1}{3}$ turbo code has a 1.5dB advantage over the rate $\frac{1}{2}$ turbo code.



(a) (b)

Figure 4.7 Rate $\frac{1}{2}$ Turbo code ExIT chart on GMM, α=0.01 (a) and 0.1 (b), pinch-off SNR limit = 3.1dB and 16.5dB.



(a) (b)

Figure 4.8 Rate $\frac{1}{3}$ Turbo code ExIT chart on GMM, α=0.01 (a) and 0.1 (b), pinch-off SNR limit = 2.7dB and 15dB.

Table 4.1 lists all pinch-off SNRs for rate $\frac{1}{2}$ and rate $\frac{1}{3}$ turbo code in different α mixtures:

α	0	0.01	0.05	0.1	0.2	0.5
R=$\frac{1}{2}$	2.2dB	3.1dB	3.8dB	16.5dB	19.5dB	21.5dB
R=$\frac{1}{3}$	1.7dB	2.7dB	3.2dB	15dB	18dB	21dB

TABLE 4.1: Comparison of the pinch-off SNR limit of rate $\frac{1}{2}$ and rate $\frac{1}{3}$ turbo codes when $0 \leq \alpha \leq 0.5$.

4.4 Upper bound on turbo code BER performance

As shown in Fig. 4.1.3, turbo codes exhibit an error floor when the probability of error performance fails to reduce rapidly at high SNR. It can be explained by approximating the BER of a turbo code, by using the union bound the average BER is bounded by [108]:

$$P_b \leq \sum_{i=1}^{2^K-1} \frac{w_i}{N} Q\left(\sqrt{\frac{2dRE_b}{N_0}}\right) \tag{4.30}$$

where w is the weight of the message sequence of the i-th message, d is the Hamming distance of a certain codeword; N is the interleaver length and R is the code rate of the turbo code. Reordering the terms corresponding to the information sequences of the same weight, equation (4.26) can be rewritten as [46]:

$$P_b \leq \sum_{i=1}^{N} \sum_{l=1}^{(N\ j)} \frac{w_i}{K} Q\left(\sqrt{\frac{2d_l RE_b}{N_0}}\right) \tag{4.31}$$

where $(N\ j)$ is the number of information sequences of weight j. As SNR increases, the first two terms of P_b dominates, so (4.27) can be approximate at high SNR as:

$$P_b \leq \sum_{j=2}^{3} \frac{jn_j}{K} Q\left(\sqrt{\frac{2d_{min}RE_b}{N_0}}\right) \tag{4.32}$$

where d_{free} denotes the minimum codeword weight among all codewords generated by the information sequences of weight j. The reason why error floors appear at high SNR can be seen from equation (4.41), that is, a smaller d_{min} will cause an error floor due to it representing low weight codewords.

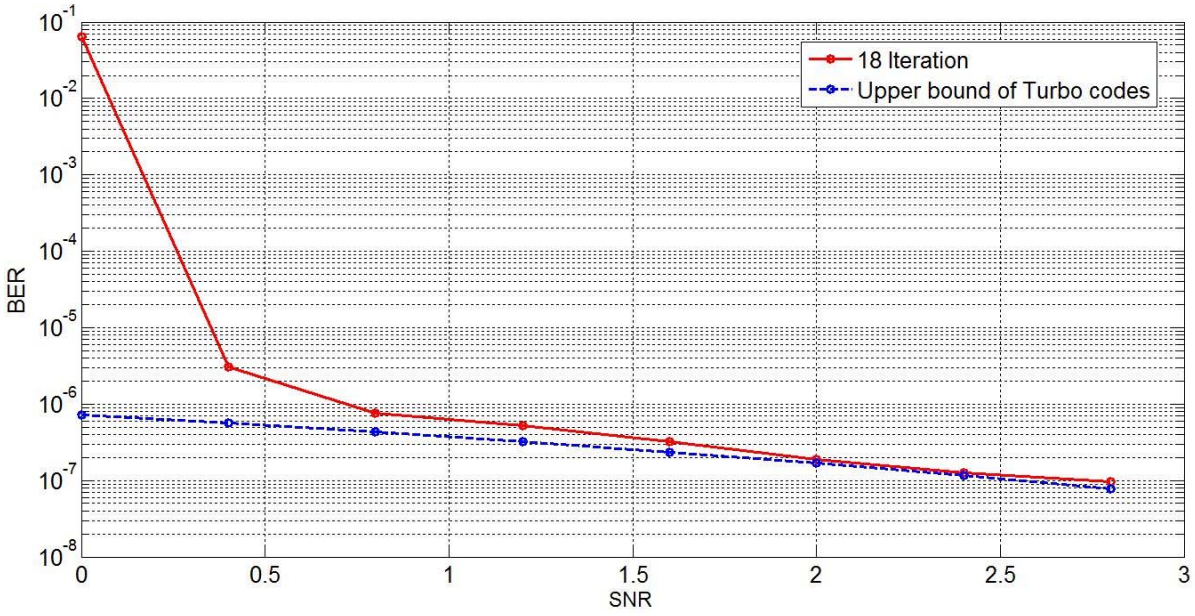


Figure 4.9 Upper bound of rate $\frac{1}{2}$ turbo code, ten decoding iterations.

Figure 4.9 shows the upper bound on the BER of a rate $\frac{1}{2}$ turbo code. It can be seen that at an SNR of 2dB, the turbo code BER converges to the upper bound and s in an error floor.

Recalling the pdf expression of GMM noise in (3.26), that related to overall noise power spectral density of GMM noise N_{GMM}:

$$N_{GMM} = (1-\alpha)N_G + \alpha N_I \tag{4.33}$$

where N_G and N_I are the noise power spectral densities for the Gaussian noise and impulsive noise respectively. By substituting (3.36) to (4.32), the asymptotic performance of a turbo code on a GMM channel can be expressed as:

$$P_{GMM} \leq \sum_{j=2}^{3} \frac{(1-\alpha)jn_j}{K} Q\left(\sqrt{\frac{2d_{min}RE_b}{N_G}}\right) + \sum_{j=2}^{3} \frac{\alpha jn_j}{K} Q\left(\sqrt{\frac{2d_{min}RE_b}{N_I}}\right) \quad (4.34)$$

Moreover, if we only take the impulsive term $\sum_{j=2}^{3} \frac{\alpha jn_j}{K} Q\left(\sqrt{\frac{2d_{min}RE_b}{N_I}}\right)$ from P_{GMM}, this gives a higher bound that actually indicates the error floor caused by impulsive noise.

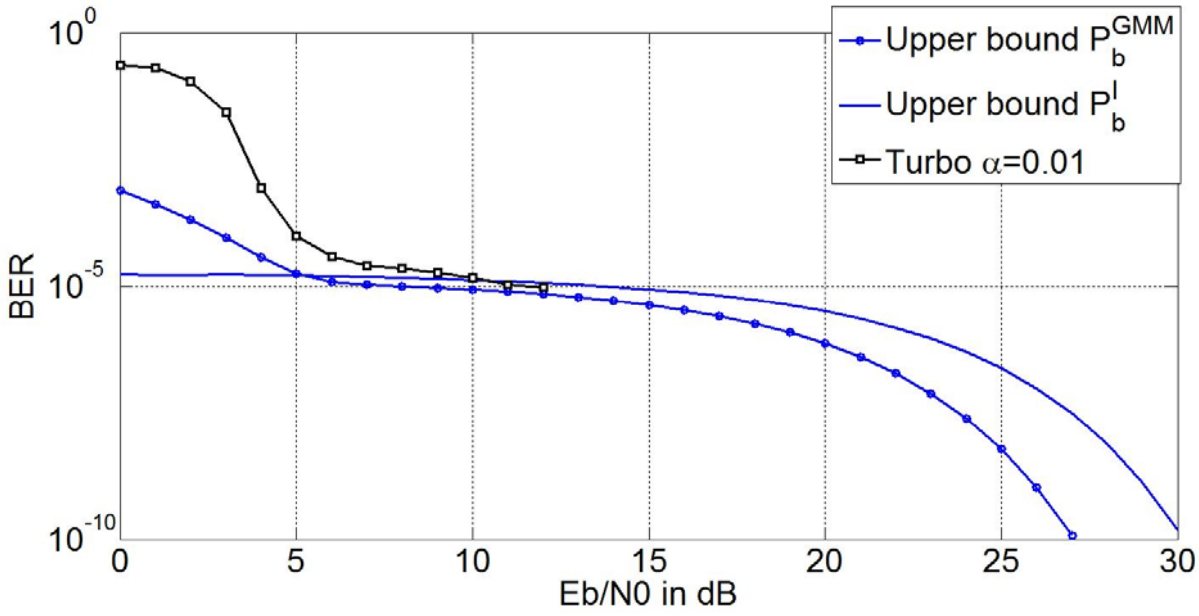


Figure 4.10 Comparison of rate $\frac{1}{2}$ turbo code with PNC at the relay, α=0.01 with upper bound and the higher impulsive bound

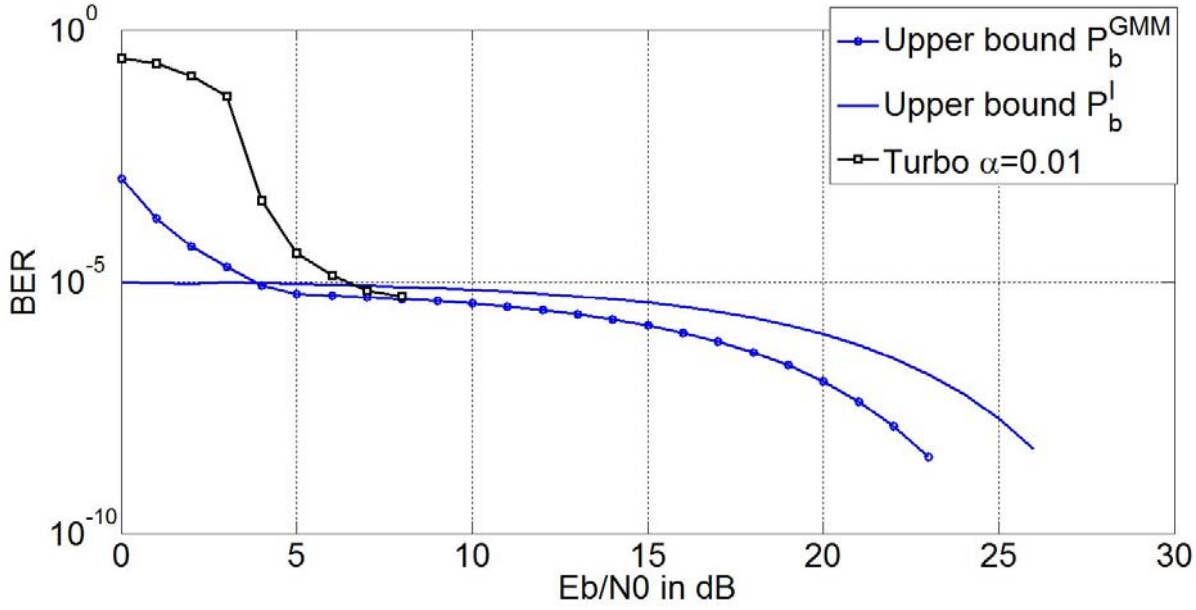


Figure 4.11 Comparison of rate $\frac{1}{3}$ turbo code on PNC at the relay, α=0.01 with upper bound and the higher impulsive bound

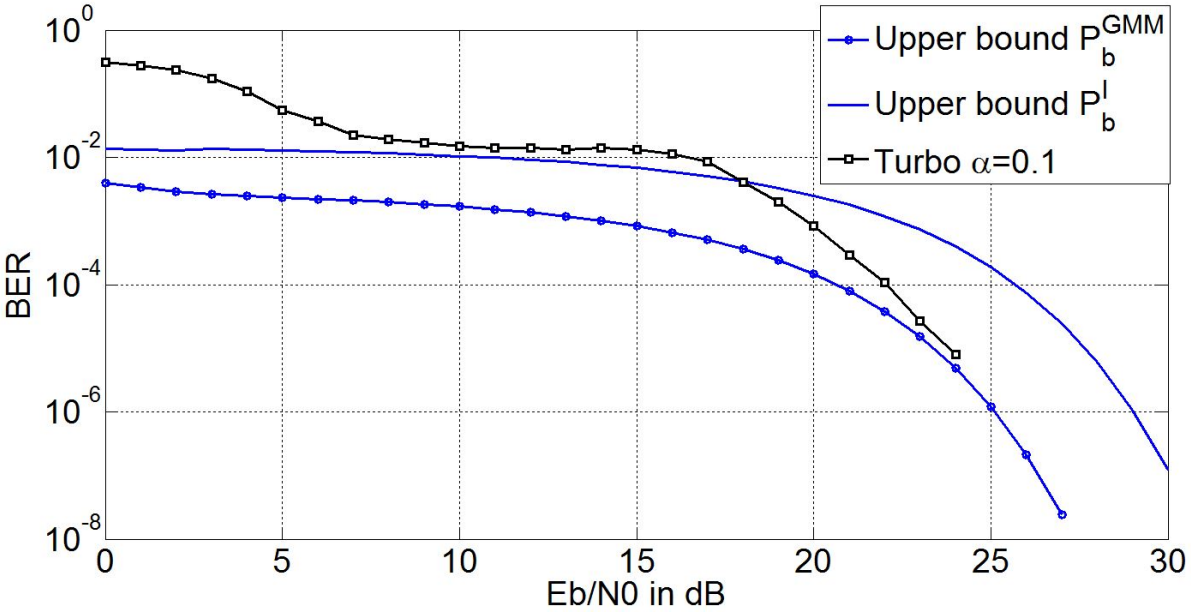


Figure 4.12 Comparison of rate $\frac{1}{2}$ turbo code on PNC at the relay, α=0.1 with upper bound and the higher impulsive upper bound

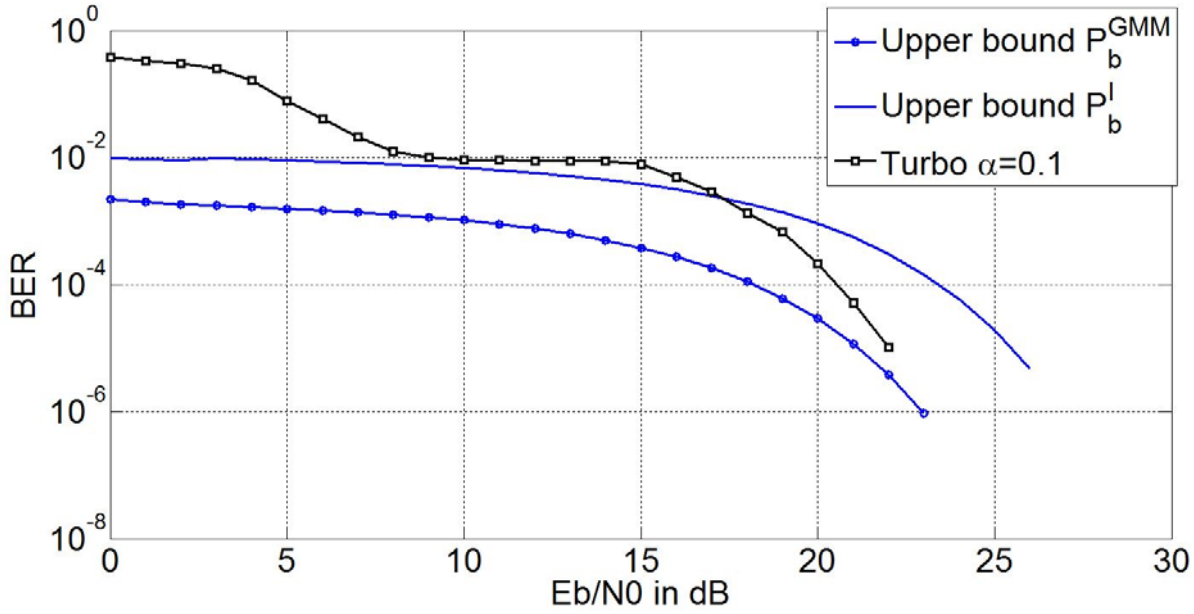


Figure 4.13 Comparison of rate $\frac{1}{3}$ turbo code on PNC at the relay, α=0.1 with upper bound and the higher impulsive bound

As SNR increases, the simulated results start to converge with the upper bound. Figure 4.10 and Figure 4.11 show the upper bounds and simulated results when α =0.01. In this case, we observe that the impulsive bound does not have a significant effect on the turbo code performance since the impulsive mixture is much lower and the error floor region is consequently much smaller, but the simulated turbo code performance converges with the upper bound at higher SNRs. In Figure 4.12 and Figure 4.13 the mixture is α =0.1 and it can be seen that, at low SNR the simulated results match closely with the higher impulsive bound and converge with the error floor region from approximately 8dB to 15dB.

4.5 BER Performance of Turbo Codes combined with PNC

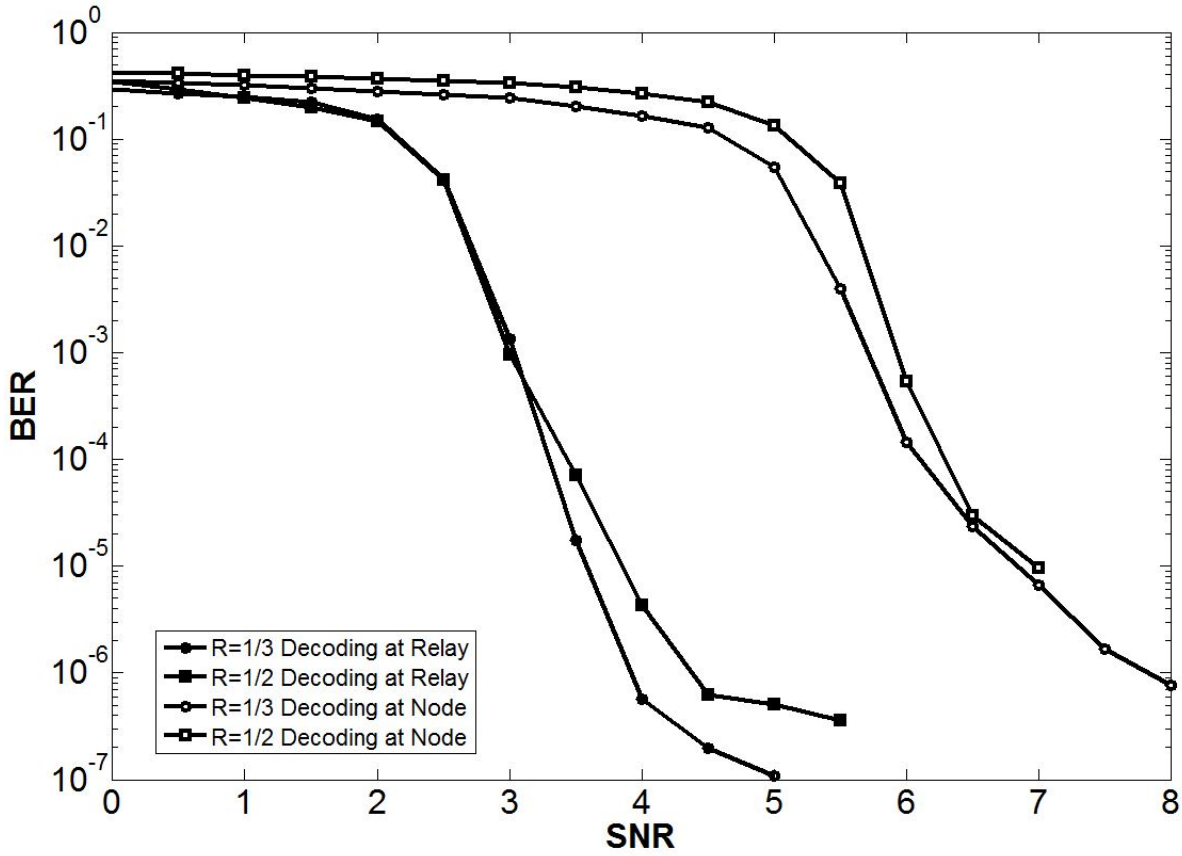


Figure 4.14 BER performance of rate $\frac{1}{2}(37,21)_8$ and rate $\frac{1}{3}(37,21)_8$ turbo codes on PNC at the relay, Interleaver length = 2,000 bits, five decoding iterations and AWGN

Figure 4.14 shows a comparison of rate $\frac{1}{2}$ and rate $\frac{1}{3}$ $(37,21)_8$ turbo codes combined with PNC, both at the relay and at a destination node. The performance of the rate $\frac{1}{2}$ punctured turbo code performs slightly worse than the rate $\frac{1}{3}$ turbo code for both at the relay and node. The waterfall regions for both codes start at around an SNR of 2dB for decoding at the relay and around 5dB for decoding at the node. It can also be seen that there is approximately a 3dB difference between decoding at relay and decoding at node for each turbo code. This is due to the codewords being decoded and encoded again at the relay before transmitting to each node, where the uncorrected errors remain in the codeword but are incorrrectly assumed to be 'correct'.

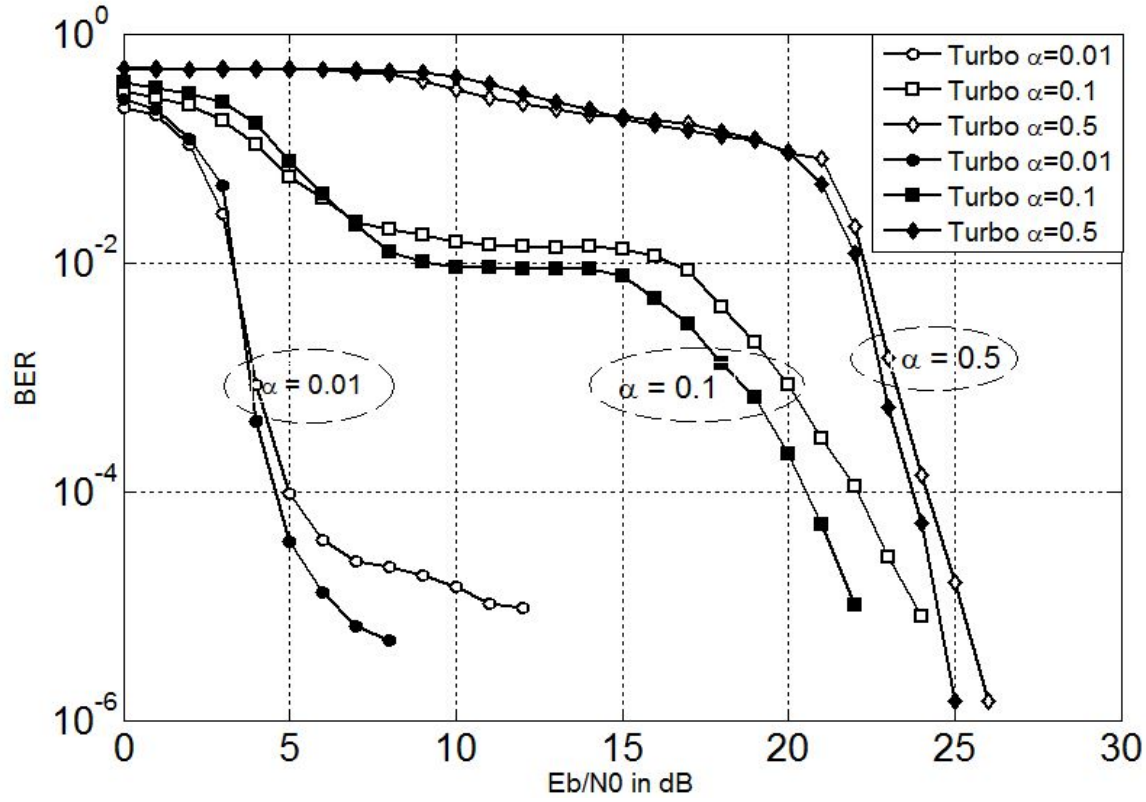


Figure 4.15 BER performance of rate $\frac{1}{2}(37,21)_8$ Turbo code (white squares) and rate $\frac{1}{3}$ Turbo code (black squares) with impulsive noise at the relay, Interleaver length = 2,000 bits, five decoding iterations

Figure 4.15 shows a comparison of rate $\frac{1}{2}$ and rate $\frac{1}{3}$ $(37,21)_8$ turbo codes combined with PNC at the relay with impulsive noise channels. At the relay, any large positive or negative impulses are clipped so that their energy is no greater than the transmitted symbol energy *E*. The performance of turbo coded PNC is seriously affected on additive impulsive noise channels resulting in error floors, as shown in the figure. The rate $\frac{1}{2}$ punctured turbo code performs slightly worse than the rate $\frac{1}{3}$ turbo code when $\alpha = 0.01$ and the waterfall regions for both codes start at around an SNR of 3dB, which is supported by the ExIT charts in Figure 4.7 and Figure 4.8. When $\alpha = 0.5$, the channel is very impulsive and both codes are having similar performance with error floor occurring at very high BERs.

4.6 PNC combined with Trellis BICM-ID

In this section, the structure of the trellis BICM-ID encoder and iterative decoders will be discussed, including the design and analysis of the trellis BICM-ID decoder suitable for combination with PNC. Trellis BICM-ID has a much lower complexity than turbo codes and is designed for fading channels, where bursts of errors occur frequently. It is well known that the performance of trellis BICM-ID on the AWGN channel is worse than turbo codes, but their performance on impulsive noise channels compared with turbo codes is not known. A BER performance comparison to the turbo code is presented with a view to seeing the difference between different network coding schemes for PNC on the impulsive channel. ExIT charts are also presented to compare the performance of PNC combined with different code rates of trellis BICM-ID code and turbo code to verify the simulation results.

4.6.1 Construction of trellis BICM-ID

The encoder of trellis BICM-ID consists of a convolutional encoder connected to an interleaver, which increases the achievable diversity order. Thus with the aid of the interleaver the code's diversity order can be extended to the binary Hamming distance of code [54]. The decoder structure of trellis BICM-ID is similar to the turbo decoder, but consists of only one soft-in-soft-out (SISO) demapper in series with a SISO decoder, in order to exchange the extrinsic information to each other to enhance the performance of the decoder. The maximum likelihood (ML) decoding of trellis BICM-ID is infeasible, due to the interleaver introducing a significant number of states. However, the trellis BICM-ID decoder has an efficient and sub-optimal decoding method that may approach the performance of the ML decoding algorithm. The basic structure of trellis BICM-ID is shown in Figure 4.16

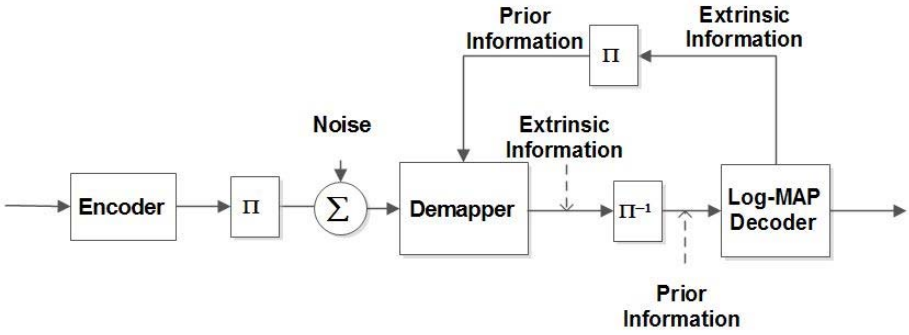


Figure 4.16 General structure of trellis BICM-ID

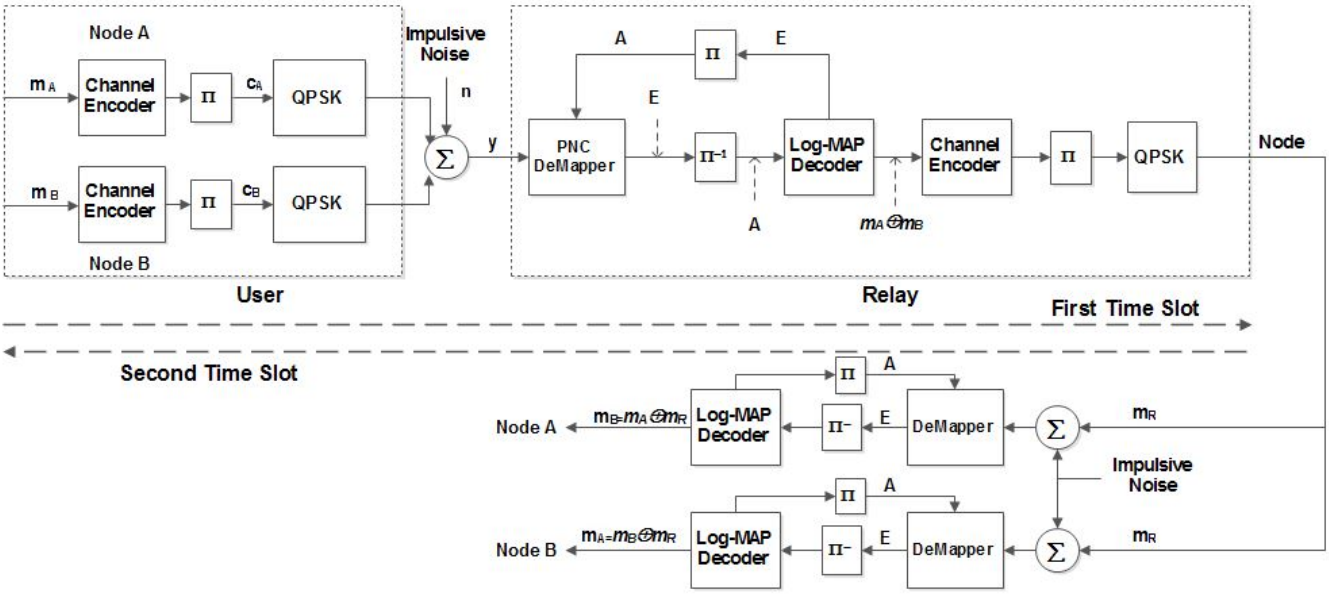


Figure 4.17 System model showing trellis BICM-ID encoder and iterative decoding processing on the PNC system.

It should be noted that trellis BICM-ID needs to use QPSK with anti-Gray mapping to ensure a coding gain. The demapper exchanges mutual information with the decoder during each decoding iteration, in order to update the LLR for more accurate demapping. The received signal *y* is demapped to form the LLR of the modulo-2 sum of the binary messages m_R and after de-interleaving the LLR is fed to the log-MAP decoder. Extrinsic information obtained from the output of the log-MAP decoder is then interleaved and fed back to the demapper, thus completing one iteration. Similarly, extrinsic information

from the demapper is de-interleaved to form the *a priori* information to the log-MAP decoder. The system model of the TWRC with PNC at the relay combined with trellis BICM-ID is shown in Figure 4.17.

4.6.2 Demapper and MAP decoder of trellis BICM-ID on PNC

Trellis BICM-ID is a spectrally efficient coded modulation scheme that has been shown to perform well under suitable signal mapping schemes with iterative demapping-decoders. In order to demap the received symbol, the conditional LLRs of the two bits, m_R^1 and m_R^2, contained in the received symbol y_R, $L(m_R^1|y)$ and $(m_R^2|y)$, can be expressed as:

$$L(m_R^1|y) = \ln\left(\frac{P(m_R^1 = 1|y)}{P(m_R^1 = 0|y)}\right) \tag{4.35}$$

and

$$L(m_R^2|y) = \ln\left(\frac{P(m_R^2 = 1|y)}{P(m_R^2 = 0|y)}\right) \tag{4.36}$$

which can be expanded as:

$$L(m_R^1|y) = \ln\left(\frac{P(m_R^1 = 1, m_R^2 = 0|y) + P(m_R^1 = 1, m_R^2 = 1|y)}{P(m_R^1 = 0, m_R^2 = 0|y) + P(m_R^1 = 0, m_R^2 = 1|y)}\right) \tag{4.37}$$

and

$$L(m_R^2|y) = \ln\left(\frac{P(m_R^1 = 0, m_R^2 = 1|y) + P(m_R^1 = 1, m_R^2 = 1|y)}{P(m_R^1 = 0, m_R^2 = 0|y) + P(m_R^1 = 1, m_R^2 = 0|y)}\right) \tag{4.38}$$

Since the coded bits are bit interleaved, we can assume they are all independent. Hence, the joint probabilities can be expressed as the product of individual probabilities:

$$P(m_R^1 = 0, m_R^2 = 1) = P(m_R^1 = 0)P(m_R^2 = 1) \tag{4.39}$$

So by applying Baye's rule:

$$P(A|B)P(B) = P(B|A)P(A) \tag{4.40}$$

the joint conditional probabilities can be expressed as:

$$P(m_R^1 = 0, m_R^2 = 1|y)P(y) = P(y|m_R^1 = 0, m_R^2 = 1)P(m_R^1 = 0, m_R^2 = 1) \quad (4.41)$$

and the LLRs of the received symbol $L(m_R^1|y)$ and $L(m_R^2|y)$ can be expanded as:

$$L(m_R^1|y) \quad (4.42)$$
$$= \ln\left(\frac{\frac{P(y|m_R^1 = 1, m_R^2 = 0)P(y|m_R^1 = 1)P(m_R^2 = 0)}{P(y)} + \frac{P(y|m_R^1 = 1, m_R^2 = 1)P(y|m_R^1 = 1)P(m_R^2 = 1)}{P(y)}}{\frac{P(y|m_R^1 = 0, m_R^2 = 0)P(y|m_R^1 = 0)P(m_R^2 = 0)}{P(y)} + \frac{P(y|m_R^1 = 0, m_R^2 = 1)P(y|m_R^1 = 0)P(m_R^2 = 1)}{P(y)}}\right)$$

Cancelling out $P(y)$ and factorising reduces $L(m_R^1|y)$ to two terms:

$$L(m_R^1|y) \quad (4.43)$$
$$= L_a(m_R^1) + \ln\left(\frac{P(y|m_R^1 = 1, m_R^2 = 0)P(m_R^2 = 0) + P(y|m_R^1 = 1, m_R^2 = 1)P(m_R^2 = 1)}{P(y|m_R^1 = 0, m_R^2 = 0)P(m_R^2 = 0) + P(y|m_R^1 = 0, m_R^2 = 1)P(m_R^2 = 1)}\right),$$

Similarly $L(m_R^2|y)$ is equal to:

$$L(m_R^2|y) \quad (4.44)$$
$$= L_a(m_R^2) + \ln\left(\frac{P(y|m_R^1 = 0, m_R^2 = 1)P(m_R^1 = 0) + P(y|m_R^1 = 1, m_R^2 = 1)P(m_R^1 = 1)}{P(y|m_R^1 = 0, m_R^2 = 0)P(m_R^1 = 0) + P(y|m_R^1 = 1, m_R^2 = 0)P(m_R^1 = 1)}\right),$$

where $L_a(m_R^1)$ is the a priori LLR of the first coded bit and $L_a(m_R^1)$ is the a priori LLR of the second coded bit, defined as:

$$L_a(m_R^1) = \ln\left(\frac{P(m_R^1 = 1)}{P(m_R^1 = 0)}\right), \quad (4.45)$$

$$L_a(m_R^2) = \ln\left(\frac{P(m_R^2 = 1)}{P(m_R^2 = 0)}\right). \quad (4.46)$$

Since

$$P(m_R^1 = 1) = 1 - P(m_R^1 = 0), \quad (4.47)$$

Therefore,

$$P(m_R^1 = 1) = \frac{e^{L_a(m_R^1)}}{1 + e^{L_a(m_R^1)}}, \tag{4.48}$$

and

$$P(m_R^1 = 0) = \frac{1}{1 + e^{L_a(m_R^1)}}. \tag{4.49}$$

Now the received LLRs can be derived as:

$$\begin{aligned} &L(m_R^1|y) \\ &= L_a(m_R^1) \\ &+ \ln\left(\frac{\frac{P(y|m_R^1=1,m_R^2=0)}{1+e^{L_a(m_R^2)}} + \frac{P(y|m_R^1=1,m_R^2=1)}{1+e^{L_a(m_R^2)}}e^{L_a(m_R^2)}}{\frac{P(y|m_R^1=0,m_R^2=0)}{1+e^{L_a(m_R^2)}} + \frac{P(y|m_R^1=0,m_R^2=1)}{1+e^{L_a(m_R^2)}}e^{L_a(m_R^2)}}\right), \end{aligned} \tag{4.50}$$

That can be simplified to:

$$\begin{aligned} L(m_R^1|y) &= L_a(m_R^1) \\ &+ \ln\left(\frac{P(y|m_R^1=1,m_R^2=0) + P(y|m_R^1=1,m_R^2=1)e^{L_a(m_R^2)}}{P(y|m_R^1=0,m_R^2=0) + P(y|m_R^1=0,m_R^2=1)e^{L_a(m_R^2)}}\right). \end{aligned} \tag{4.51}$$

and

$$\begin{aligned} L(m_R^2|y) &= L_a(m_R^2) \\ &+ \ln\left(\frac{P(y|m_R^1=0,m_R^2=1) + P(y|m_R^1=1,m_R^2=1)e^{L_a(m_R^1)}}{P(y|m_R^1=0,m_R^2=0) + P(y|m_R^1=1,m_R^2=0)e^{L_a(m_R^1)}}\right). \end{aligned} \tag{4.52}$$

From Figure 4.6, we know the received symbol belongs to nine possible complex values at the relay. By following the constellation diagram the conditional pdfs can be expressed as:

$$p(y|m_R^1 = 0, m_R^2 = 0)$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(e^{-\frac{\left[(y_I-2E)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I-2E)^2+(y_Q+2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I+2E)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I+2E)^2+(y_Q+2E)^2\right]}{2\sigma^2}} \right) dy_I dy_Q \quad (4.53)$$

$$p(y|m_R^1 = 1, m_R^2 = 1)$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(2e^{-\frac{\left[(y_I-2E)^2+(y_Q)^2\right]}{2\sigma^2}} + 2e^{-\frac{\left[(y_I+2E)^2+(y_Q)^2\right]}{2\sigma^2}} \right) dy_I dy_Q \quad (4.54)$$

$$p(y|m_R^1 = 0, m_R^2 = 1)$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(2e^{-\frac{\left[(y_I)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + 2e^{-\frac{\left[(y_I)^2+(y_Q+2E)^2\right]}{2\sigma^2}} \right) dy_I dy_Q \quad (4.55)$$

$$p(y|m_R^1 = 1, m_R^2 = 0) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} 4e^{-\frac{(y_I^2+y_Q^2)}{2\sigma^2}} dy_I dy_Q \quad (4.56)$$

By substituting the conditional probability into $L(m_R^1|y)$ and $L(m_R^2|y)$, the extension of $L(m_R^1|y)$ and $L(m_R^2|y)$ can be viewed in the appendix.

4.7 ExIT chart of trellis BICM-ID

The ExIT-chart technique for trellis BICM-ID is quite similar to the turbo decoder. Let us denote the decoder to demapper message as $M_k^{dec-dem}$ and the demapper to decoder message is $M_k^{dem-dec}$. Also we define the two symbols M_e and M_A as the average mutual information between the coded bit X_k, where $k = 1, \dots, n$ and n is the length of message and the decoder to demapper message $M_k^{dec-dem}$ and $M_k^{dem-dec}$. So in general the relationship between the extrinsic information and *a priori* information in the trellis BICM-ID decoder can be expressed as:

$$M_E = \frac{1}{n}\sum_{k=1}^{n} I\left(X_k; M_k^{dec-dem}\right), \tag{4.57}$$

$$M_A = \frac{1}{n}\sum_{k=1}^{n} I\left(X_k; M_k^{dem-dec}\right), \tag{4.58}$$

where M_E and M_A correspond to the extrinsic information from the output of the decoder to the input of the demapper, and vice versa. M_E and M_A has the relationship:

$$M_A = exit_{dem}(M_E), \tag{4.59}$$

$$M_E = exit_{dec}(M_A). \tag{4.60}$$

This relationship represents the transfer function of the extrinsic information and *a priori* information exchange between the decoder and demapper in the trellis BICM-ID decoder. The ExIT charts of trellis BICM-ID at the relay on GMM impulsive noise channels is shown in Figures 4.18 – 4.21. When performing the ExIT chart analysis, we chose values of the GMM mixture constant α from 0.01 to 0.1 to analyse the convergence behaviour of the turbo codes at the relay of the conventional two-way relay communications. From Figure. 4.18 it can be seen that the lowest SNR, or pinch-off SNR limit, where the SNR of the rate $\frac{1}{2}$ trellis BICM-ID converges is 6.5dB. Similarly, Figure.4.19 shows the ExIT charts for the rate $\frac{1}{3}$ trellis BICM-ID and we observe that the pinch-off SNR limit is 5.6dB, which is a 0.9dB advantage over the rate $\frac{1}{2}$ trellis BICM-ID.

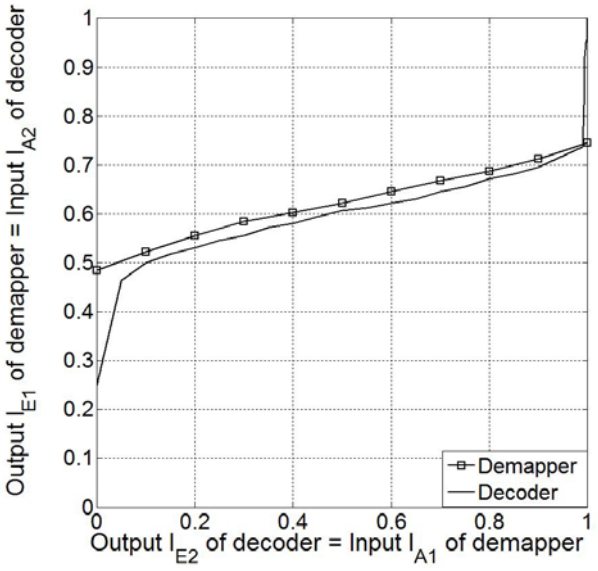


Figure 4.18 Rate $\frac{1}{2}$ trellis BICM-ID ExIT chart on PNC over GMM, α=0.01 pinch-off SNR limit=6.5dB.

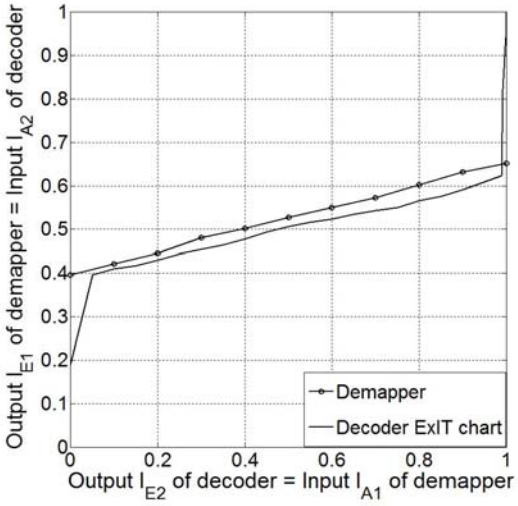


Figure 4.19 Rate $\frac{1}{3}$ trellis BICM-ID ExIT chart on PNC over GMM, α=0.01 pinch-off SNR limit=5.6dB.

Figure. 4.20 shows the ExIT charts of the rate $\frac{1}{2}$ trellis BICM-ID when α=0.1. In this case, the pinch-off SNR limit of the rate $\frac{1}{2}$ trellis BICM-ID is 10.5dB; we also see this behaviour in the ExIT charts of the rate $\frac{1}{3}$ trellis BICM-ID is shown in Figure.4.21, where the rate $\frac{1}{3}$ trellis BICM-ID has a 1dB advantage over the rate $\frac{1}{2}$ trellis BICM-ID.

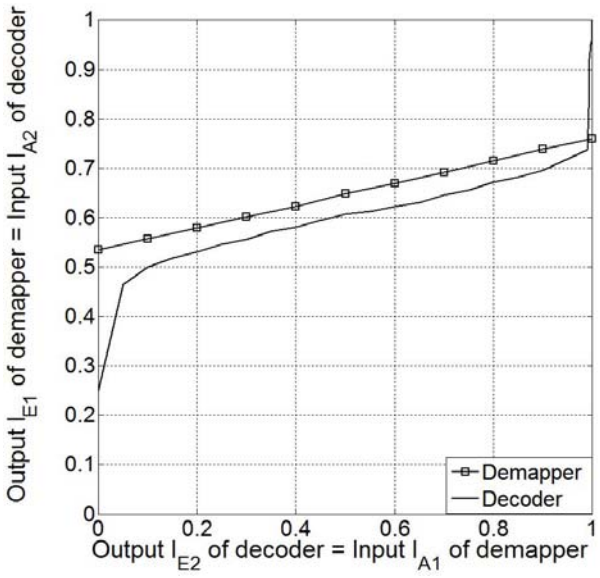


Figure 4.20 Rate $\frac{1}{2}$ trellis BICM-ID ExIT chart on PNC over GMM, α=0.1 pinch-off SNR limit=10.5dB.

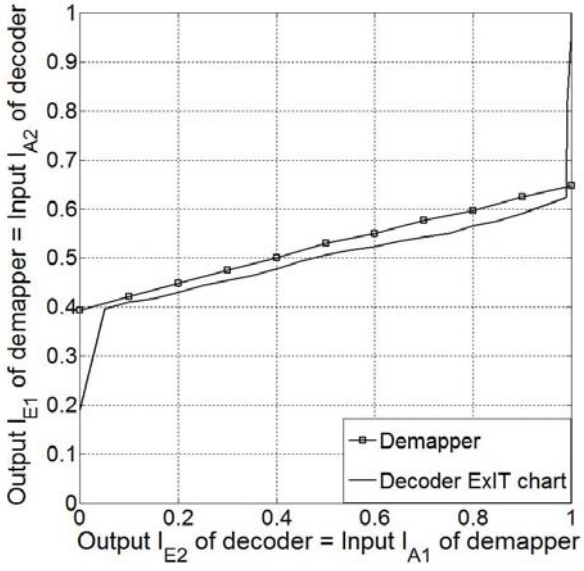


Figure 4.21 Rate $\frac{1}{3}$ trellis BICM-ID ExIT chart on PNC over GMM, α=0.1 pinch-off SNR limit=9.5dB.

4.8 Performance of Trellis BICM-ID

After performing the ExIT chart analysis of trellis BICM-ID at the relay on GMM impulsive noise channels in the previous section, we now compare simulation results of trellis BICM-ID to see the advantage of trellis BICM-ID. Earlier in the chapter, lower bounds on the turbo code error floor were presented to validate simulation results. However, this is not possible for trellis BICM-ID since they do not exhibit error floors. Therefore, only the ExIT charts in the previous section can be used to validate the simulation results.

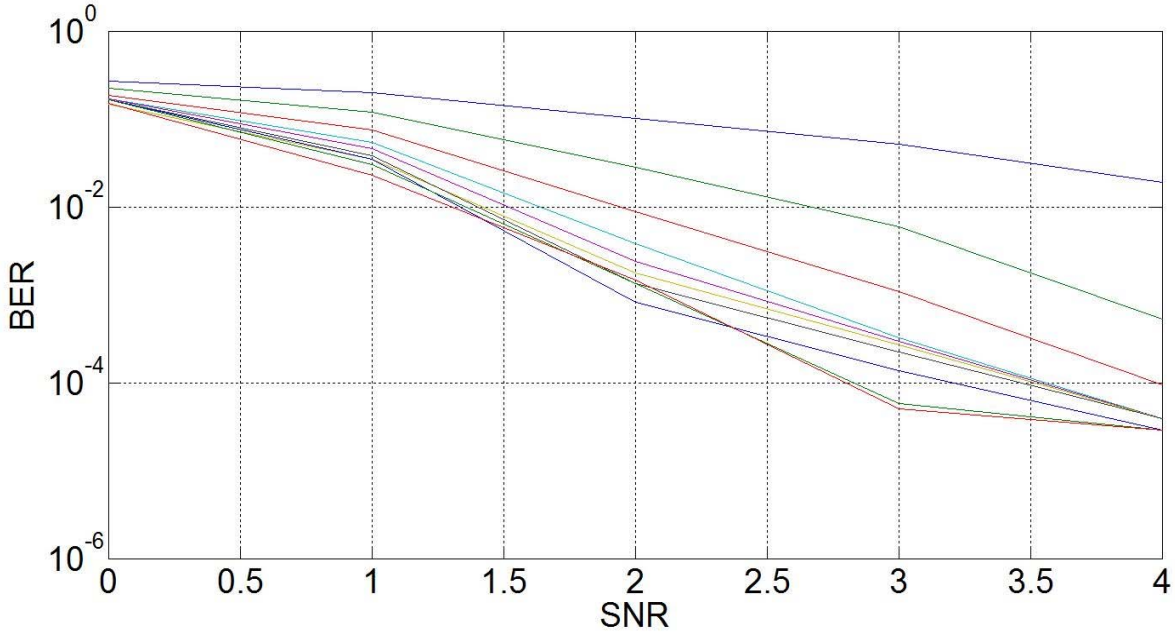


Figure 4.22 Simulation BER of rate $\frac{1}{2}$ $(7,5)_8$ trellis BICM-ID in point to point transmission on AWGN channel, ten iterations.

Figure 4.22 shows the simulated BER of a rate $\frac{1}{2}$ trellis BICM-ID in point to point transmission for ten decoding iterations. It can be seen that the performance of trellis BICM-ID improves as the number of decoding iteration increases. After ten iterations the decoder reaches an error floor at the BER of 10^{-4} at the SNR of 3dB.

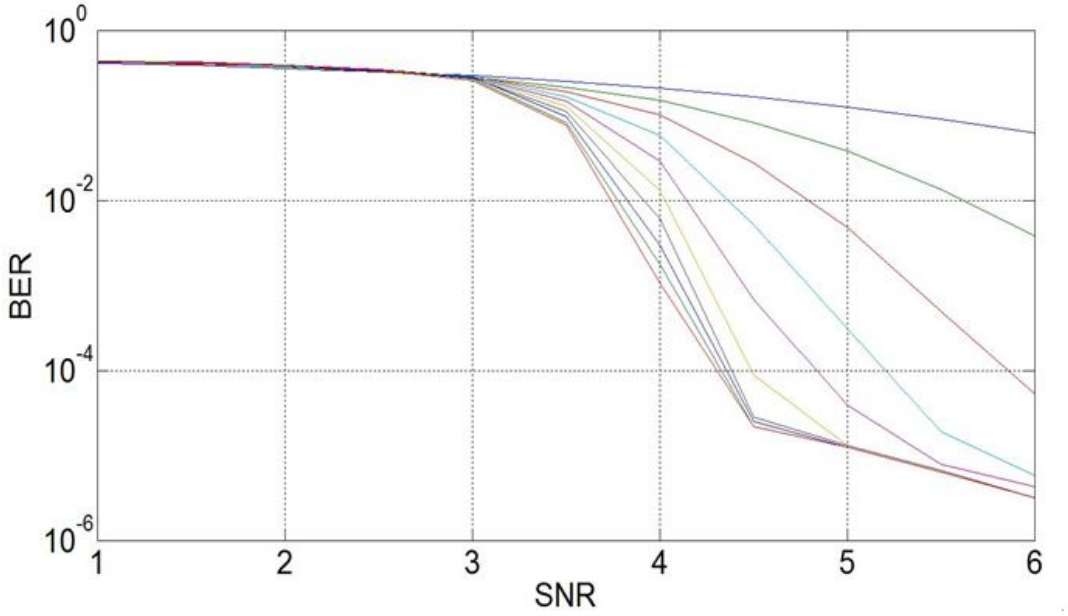


Figure 4.23 BER performance of rate $\frac{1}{3}$ trellis BICM-ID on PNC over AWGN channel at the relay with ten iterations.

In Figure 4.23 the BER performance of PNC combined with rate $\frac{1}{3}$ trellis BICM-ID at the relay on AWGN channels is shown for ten iterations. Compared to the previous BER of trellis BICM-ID on AWGN channel without PNC in Figure 4.22, there is an approximately 2dB degradation. The 'waterfall' region starts around 3.5dB, and the error floor occurs around 4.5dB.

Figure 4.24 shows a comparison of PNC combined with rate $\frac{1}{2}$ and $\frac{1}{3}$ trellis BICM-ID on AWGN channels both at the relay and node. Clearly rate $\frac{1}{3}$ trellis BICM-ID outperforms rate $\frac{1}{2}$ trellis BICM-ID due to the extra redundancy and increased minimum Hamming distance.

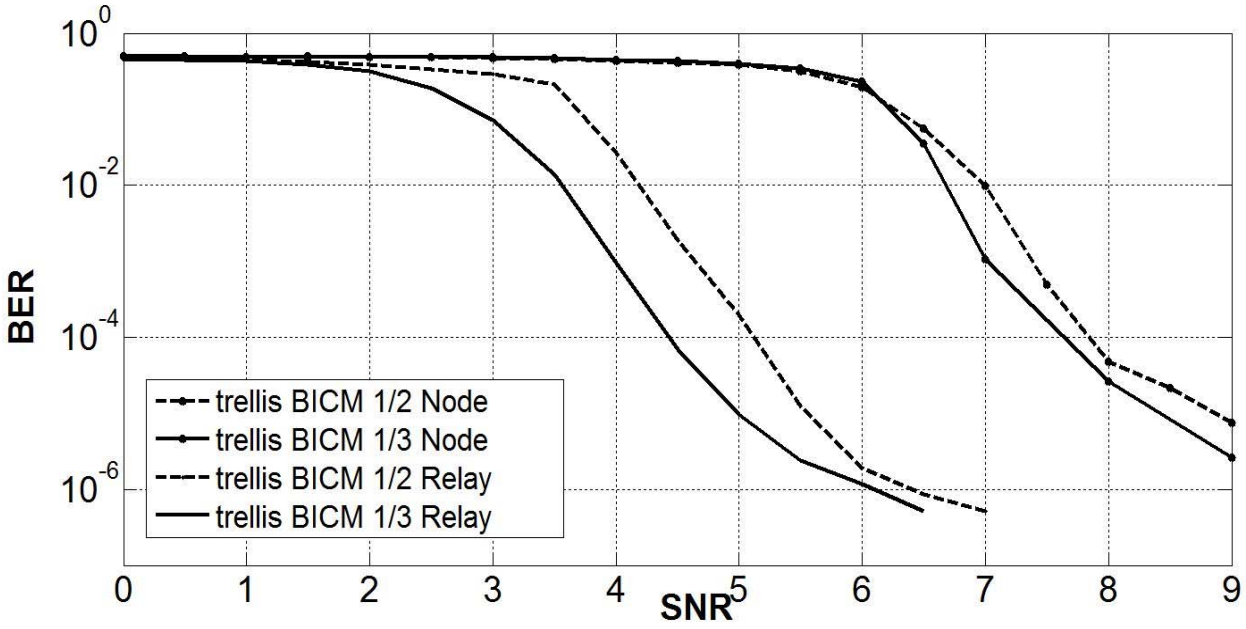


Figure 4.24 BER performance of rate $\frac{1}{2}$ and $\frac{1}{3}$ trellis BICM-ID on PNC over AWGN channel both at the relay and destination nodes.

In Figure 4.25 – 4.26, a comparison of PNC combined with rate $\frac{1}{2}$ and $\frac{1}{3}$ trellis BICM-ID on GMM impulsive noise channels with the different mixture is presented. As before, the rate $\frac{1}{3}$ trellis BICM-ID has better performance than the rate $\frac{1}{2}$ trellis BICM-ID, but the coding gain is even larger than on the AWGN channel, which is approximately 4.5 dB. By increasing α the performance of trellis BICM-ID becomes worse due to the impulsiveness of the channel increasing. Compared to the performance of trellis BICM-ID on the AWGN channel, even a small mixture of impulsive noise will degrade the BER significantly. e.g. when $\alpha = 0.01$, the degradation in BER is nearly 3dB worse. As the impulsiveness of the noise grows larger to $\alpha = 0.5$, the BER of trellis BICM-ID suffers a huge error floor at lower SNR.

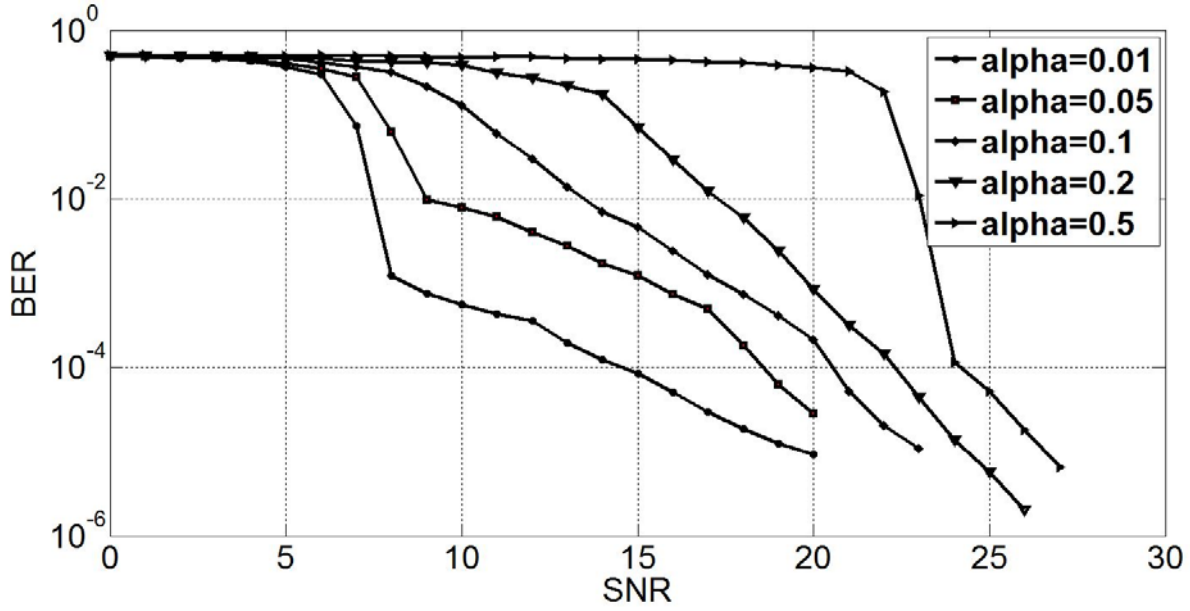


Figure 4.25 BER performance of rate $\frac{1}{2}$ trellis BICM-ID on PNC at the relay over GMM channel with different GMM mixture constant α.

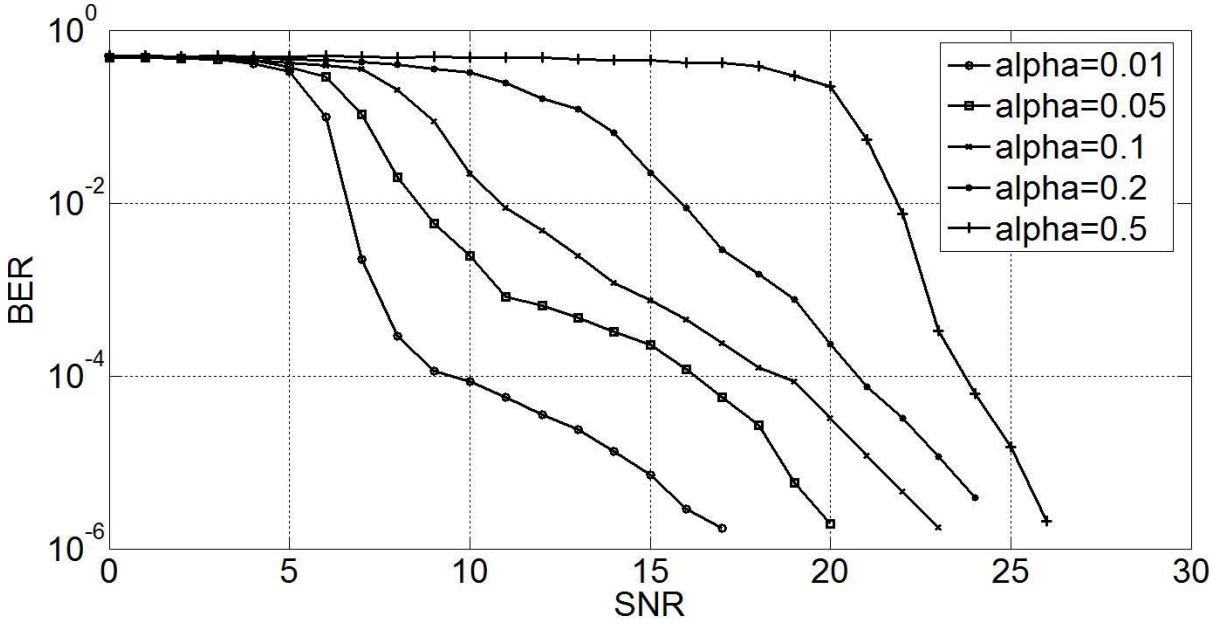


Figure 4.26 BER performance of rate $\frac{1}{3}$ trellis BICM-ID on PNC at the relay over GMM channel with different GMM mixture constant α.

Table 4.2 lists all pinch-off SNR for rate $\frac{1}{2}$ and rate $\frac{1}{3}$ trellis BICM-ID at the relay over GMM channels for different α mixtures:

α	0	0.01	0.05	0.1	0.2	0.5
R=$\frac{1}{2}$	3.5dB	6.5dB	7.5dB	10.5dB	16.5dB	21.5dB
R=$\frac{1}{3}$	3dB	5.6dB	6.5dB	9.5dB	15dB	20.5dB

TABLE 4.2: Comparison of the pinch-off SNR limit between rate $\frac{1}{2}$ and rate $\frac{1}{3}$ trellis BICM-ID when $0 \leq \alpha \leq 0.5$.

4.9 Performance comparison of Turbo code and Trellis BICM-ID on PNC over impulsive noise channel

We now comparing difference between two channel coding schemes and discuss their advantages and disadvantages when combined with PNC.

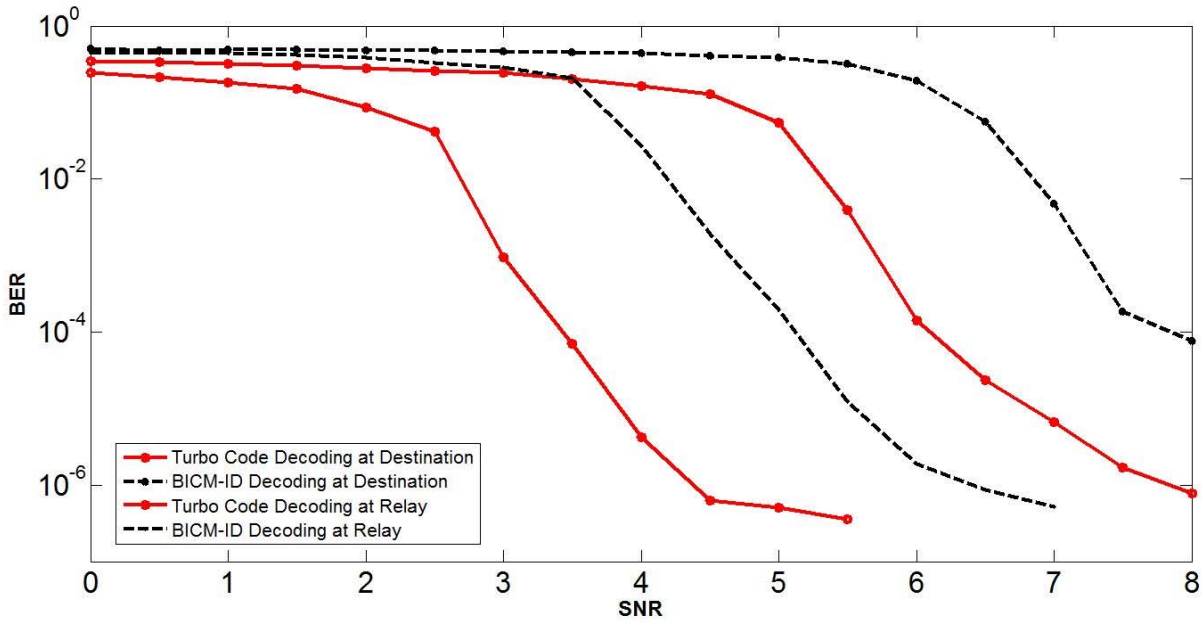


Figure 4.27 BER performances comparison of rate $\frac{1}{2}$ turbo code and rate $\frac{1}{2}$ trellis BICM-ID on PNC over AWGN channel, decoding at relay/destination node.

In Fig 4.27-4.28, the comparison of rate $\frac{1}{2}$ and rate $\frac{1}{3}$ channel codes have been made. It can be seen that turbo code outperforms trellis BICM-ID under all conditions with approximately 1.5dB coding gain for both decoding at relay/destination. There is no doubt that turbo codes are a good option for PNC on AWGN channels. However, referencing the pinch-off SNR of both coding schemes, it can be seen that the trellis BICM-ID codes have advantages when combined with PNC on impulsive noise channels when $0.1 \leq \alpha \leq 0.5$, as shown in TABLE 4.3 and verified in Figures 4.29 - 4.30.

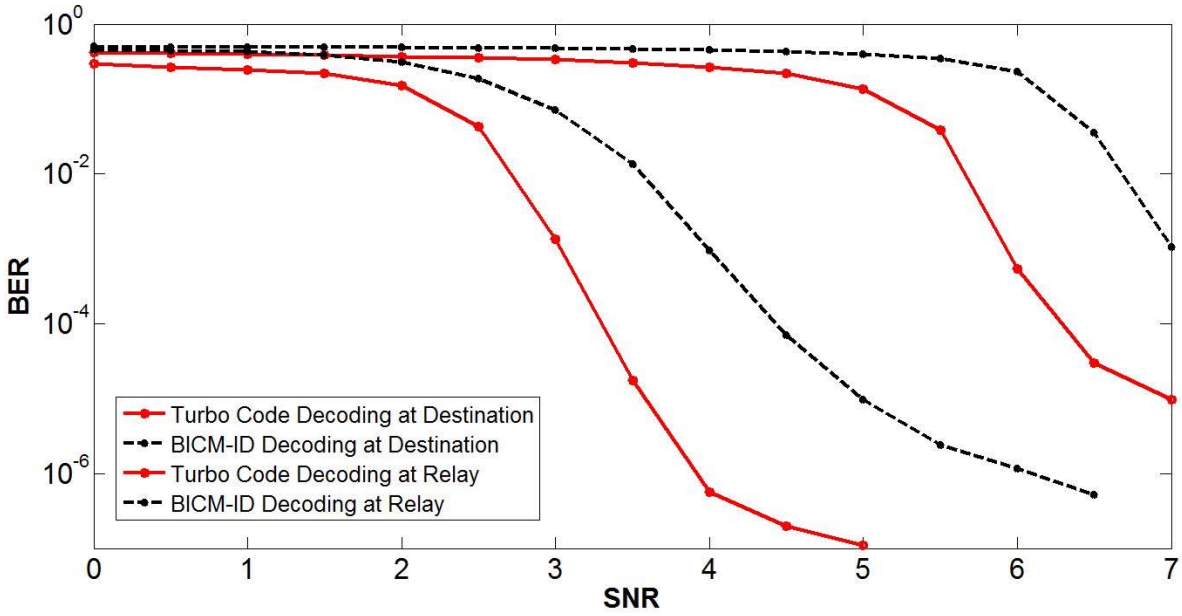


Figure 4.28 BER performances comparison of rate $\frac{1}{3}$ turbo code and rate $\frac{1}{3}$ trellis BICM-ID on PNC over AWGN channel, decoding at relay/destination.

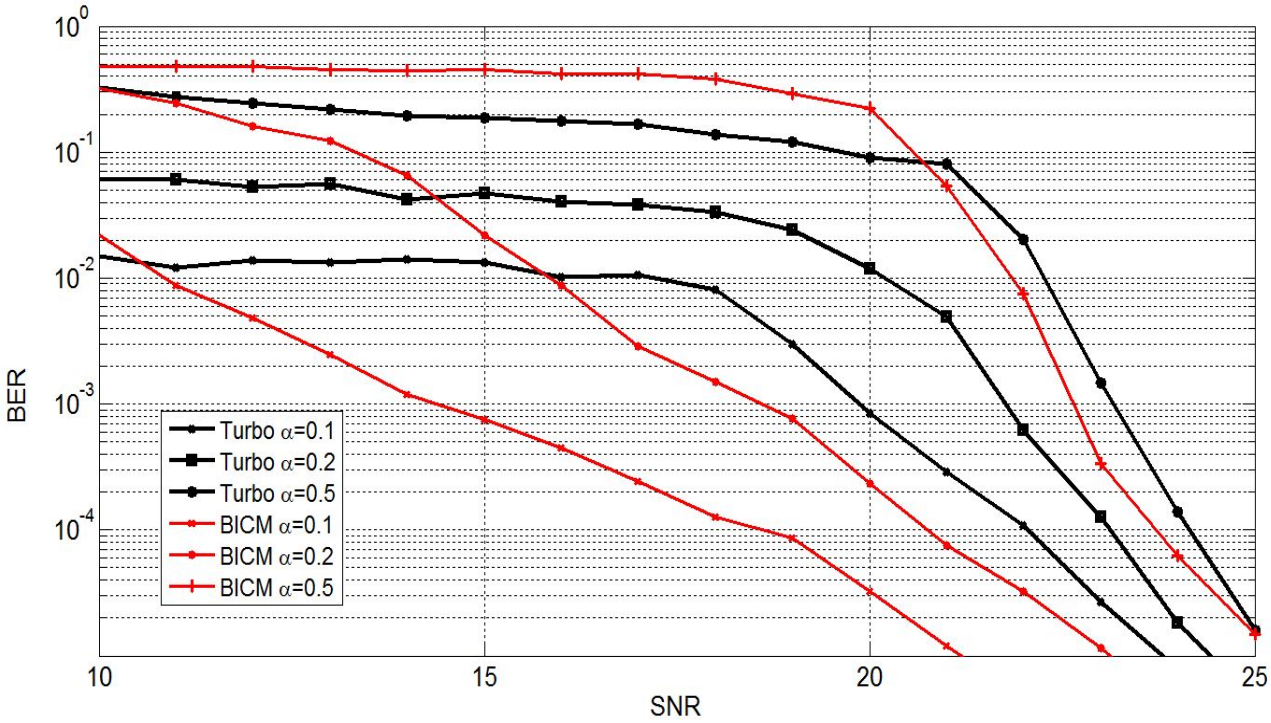


Figure 4.29 Comparison of rate $\frac{1}{2}$ turbo codes and trellis BICM-ID when $0.1 \leq \alpha \leq 0.5$.

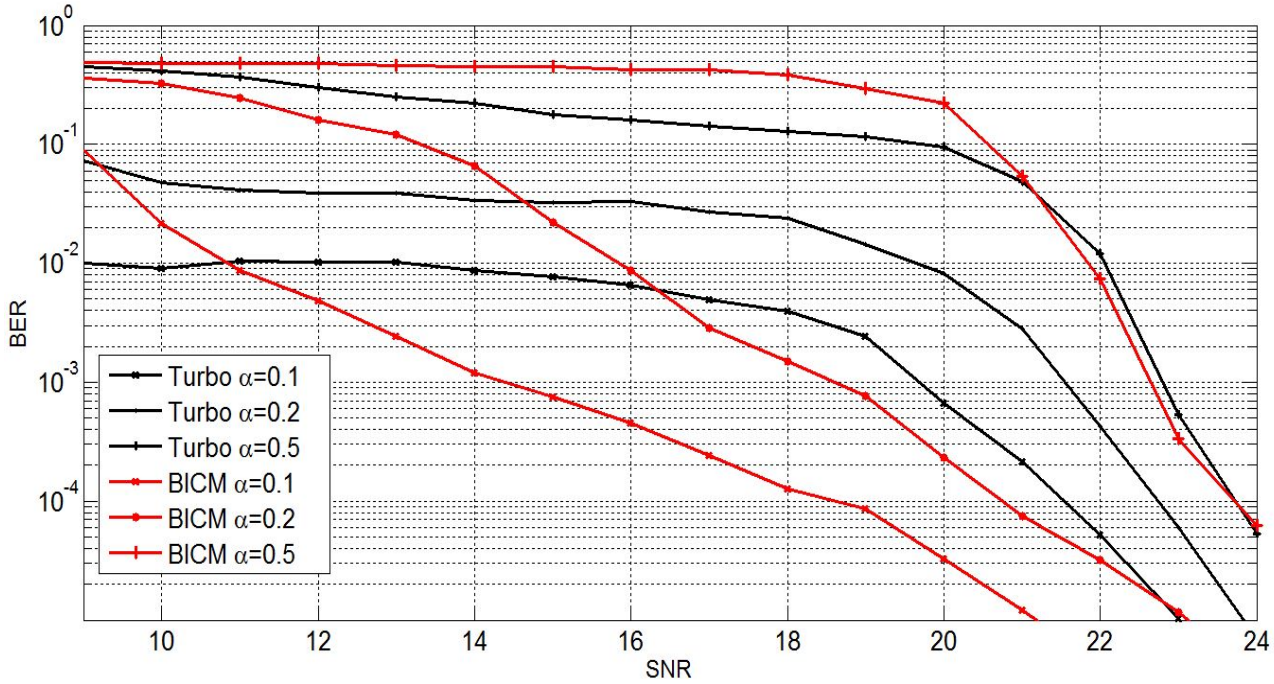


Figure 4.30 Comparison of rate $\frac{1}{3}$ turbo codes and trellis BICM-ID when $0.1 \leq \alpha \leq 0.5$.

Turbo Code				Trellis BICM-ID			
α	0.1	0.2	0.5	α	0.1	0.2	0.5
R=$\frac{1}{2}$	16.5dB	19.5dB	21.5dB	R=$\frac{1}{2}$	10.5dB	16.5dB	21.5dB
R=$\frac{1}{3}$	15dB	18dB	21dB	R=$\frac{1}{3}$	9.5dB	15dB	20.5dB

TABLE 4.3: Comparison of the pinch-off SNR limit between rate $\frac{1}{2}$ and rate $\frac{1}{3}$ turbo codes and trellis BICM-ID when $0.1 \leq \alpha \leq 0.5$.

From Figs 4.29 - 4.30, it can be seen that when the impulsive noise mixture varies with $0.1 \leq \alpha \leq 0.5$, trellis BICM-ID outperforms turbo codes especially in the range $0.1 \leq \alpha \leq 0.2$. The turbo codes occurred with error floors at lower SNR, and the level of error floors are even higher when the channel is more impulsive, $\alpha = 0.1$ at BER of 10^{-2} and $\alpha = 0.2$ at BER of 10^{-1}. However, trellis BICM-ID has avoided that effect and starts to fully decode the signals at much lower SNR than turbo codes. Overall, the comparisons of the BER performance of both channel coding schemes on PNC over impulsive channel have verified our pinch-off SNR results, and trellis BICM-ID has shown its advantages when combined with PNC on impulsive noise channels.

4.10 Conclusions

In this chapter, an analysis of rate $\frac{1}{2}$ and rate $\frac{1}{3}$ turbo codes and trellis BICM-ID combined with channel coded PNC on AWGN channels and additive impulsive noise channels has been investigated on a conventional TWRC. We have shown that the performance of turbo codes is severely affected on the GMM noise channel compared to AWGN channels, especially when the mixture is high, and ExIT charts have been presented to show the convergence behaviour of the turbo decoder for different mixtures of impulsiveness. The pinch-off SNR values for rate $\frac{1}{2}$ and $\frac{1}{3}$ turbo codes have also been determined from the ExIT charts and match closely with the simulation results. The error floors caused by impulsive noise are analysed by determining the upper bound on the

performance of turbo codes at the relay. We observed that for higher mixtures the error floor is determined solely by the impulses of the GMM noise channel at low SNRs, but this effect quickly reduces with increasing SNR and the Gaussian noise part of the GMM noise channel has more of an effect on the performance.

Compared to turbo coded PNC on the impulsive channel, when the impulsiveness is low, i.e. $\alpha \leq 0.1$, trellis BICM-ID perform worse than the turbo codes with a degradation of 2.5dB. This is because at the lower SNR region, the turbo codes could overcome the interferences from the impulsive by interleaving the information bits that input to the second encoder, so that few errors can be corrected during the iterative decoding process. However, when $0.1 \leq \alpha \leq 0.5$, trellis BICM-ID starts to achieve a better performance than turbo codes. When the impulsiveness is higher than a threshold, the iterative decoders cannot correct the error in the received signal, which the errors remain in the loop not matter how many iterations the decoder take.

Trellis BICM-ID performs better than the turbo code when $0.1 \leq \alpha \leq 0.5$ because the impulsiveness in the channel is more severe, resulting in very large LLR magnitudes. In the turbo decoder, it is difficult for each component decoder to correct this large LLR value meaning it will be passed on to the other component decoder, affecting other LLR values and causing an error floor. However, trellis BICM-ID is less affected because the decoder and demapper are independent of each other. The effect of this can be seen more clearly in the EXIT charts in figs. 4.20 and 4.21. We observe that, because the two curves are not symmetrical, the BICM-ID receiver converges at lower SNRs than the turbo decoder when $\alpha = 0.1$. When $\alpha = 0.01$, the impulsiveness is lighter and the turbo decoder copes better with the less frequent larger impulses. This can be seen in the EXIT charts of figs. 4.7(a) and 4.8(a), where the turbo decoder converges at lower SNRs than trellis BICM-ID.

Overall, the iterative decoding behavior of trellis BICM-ID combined with PNC on impulsive channels at different α has been analyzed and compared with similar turbo codes in the same scenario. The ExIT charts of trellis BICM-ID at the relay with two values of α were presented. We observed that the SNR values that signify the start of the 'waterfall regions' of the BER curves also matches closely with the pinch-off SNRs found from the ExIT charts. There are many environments where impulsive noise is present, such as interference from nearby machinery or power line communications,

where the noise at the receiver is non-Gaussian. The interference due to the two users' incoming signals at the relay causes degradation in performance compared to a single user, uni-directional coded system.

We have shown that both turbo code and trellis BICM-ID are good choices for a TWRC with PNC on impulsive noise channels and present a good trade-off between performance, spectral efficiency and complexity. There is considerable scope for further research in this area to achieve further improvements in performance by investigating new signal processing techniques and channel code design methodologies specifically for impulsive noise channels.

Chapter 5

Physical Layer Network Coding combined with Non-Binary Convolutional Codes

5.1 Introduction

Physical layer network coding (PNC) combined with binary error-correcting codes is commonly employed in the literature. In the previous chapter, PNC was combined with binary turbo codes and trellis BICM, where the focus was on the performance of these schemes on additive impulsive noise channels. However, the use of non-binary codes is less common in the literature and it seems that the combination of PNC with non-binary codes has not been investigated previously. Therefore, the major novelty in this chapter is the performance analysis of PNC combined with non-binary convolutional code on additive impulsive noise channels.

The chapter begins by defining the encoder structure and Maximum A Posteriori (MAP) decoding algorithm for a non-binary convolutional code defined in finite fields. This is followed by a theoretical performance analysis of PNC combined with non-binary codes on additive impulsive noise channels, which are modelled using the Gaussian mixture model as in the previous chapter. A union bound on the BER performance is derived and simulation results are also presented to validate the analysis. Finally, the design parameters for non-binary convolutional codes on impulsive noise channels are investigated to optimize their performance when combined with PNC.

5.2. Q-ary Convolutional Code Encoder

The structure of a q-ary convolutional codes encoder is shown in Figure 5.1.

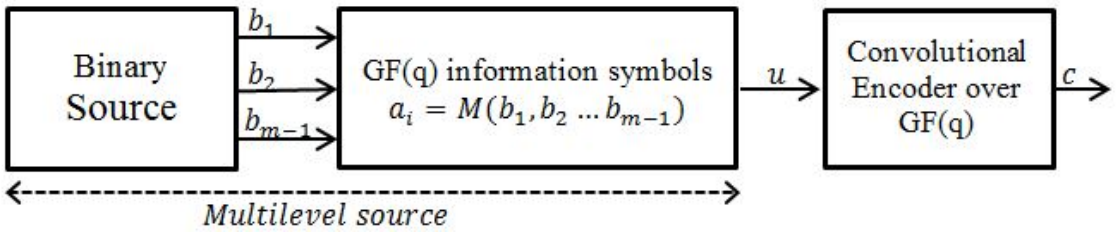


Figure 5.1 Encoder structures of q-ary convolutional codes

As shown in Figure 5.1, the encoder consists of three parts: the first two parts involve the multilevel source mapping of binary bits $b_1 \ldots b_{m-1}$ to the finite field symbols a_i in GF(q). These symbols are encoded by a convolutional encoder to generate the codeword defined in GF(q).

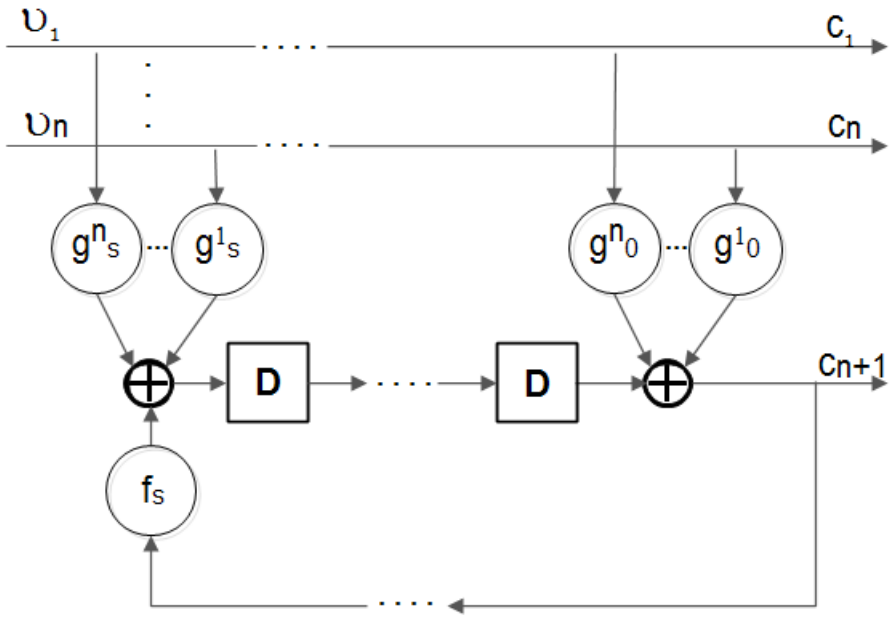


Figure 5.2 General encoder of q-ary convolutional codes.

In Fig 5.2, a general encoder of q-ary convolutional codes is given that the input symbols $\upsilon_1 \ldots \upsilon_n \in [0,1,\beta,\beta^2]$ to obtain the output of $c_1 \ldots c_n$. A simple convolutional encoder over GF(4) is shown in Figure 5.3, which has a rate of $\frac{1}{2}$ and is denoted as the $\beta\beta^2/1$ 4-ary convolutional code.

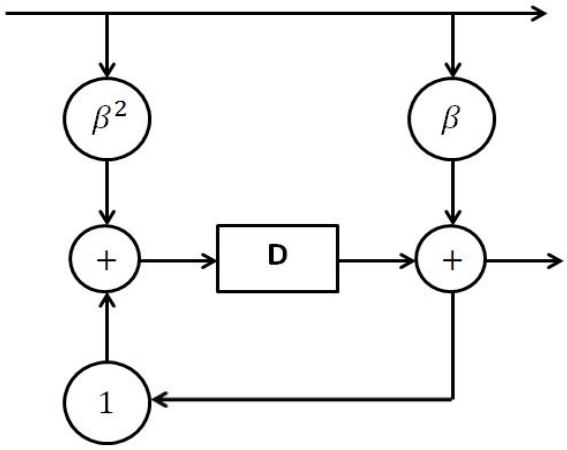


Figure 5.3 Convolutional encoder of rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code

It consists of one memory element with two feed forward coefficients $\beta\beta^2$, and one feedback coefficient 1. The state table for this rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code is shown in Table 5.1.

Input	Initial State	Next State	Output
0	0	0	00
1	0	1	1 β
β	0	β	β0
β^2	0	β^2	$\beta^2$1
0	1	1	01

1	1	0	$1\beta^2$
β	1	β^2	$\beta\ \beta$
β^2	1	β	$\beta^2 0$
0	β	β^2	$0\ \beta$
1	β	β	10
β	β	1	$\beta 1$
β^2	β	0	$\beta^2\beta^2$
0	β^2	β^2	$0\ \beta^2$
1	β^2	β	11
β	β^2	1	$\beta\beta^2$
β^2	β^2	0	$\beta^2\beta$

Table 5.1. State table for rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code

Furthermore, the corresponding trellis diagram is shown in Figure 5.4 Comparing to the binary trellis, clearly we can see that although the number of states for the 4-ary convolutional code is the same as the $(7, 5)_8$ binary convolutional code, the number of branches in the trellis are doubled since there are now four branches entering and leaving each node. From the state table, we can also draw the signal-flow graph as shown in Figure 5.5, which is more complicated than the signal flow graph of a 4-state binary convolutional code.

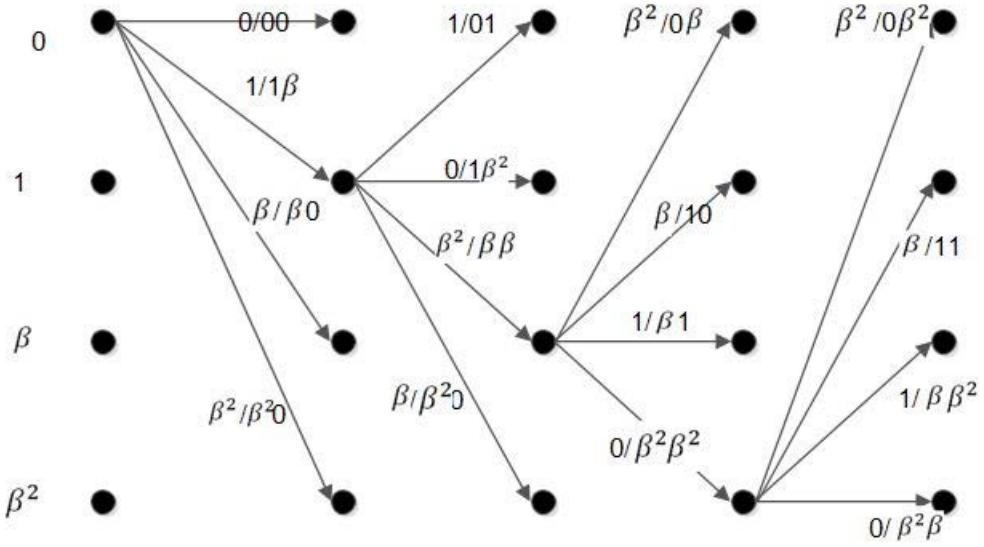


Figure 5.4 Trellis of a rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code.

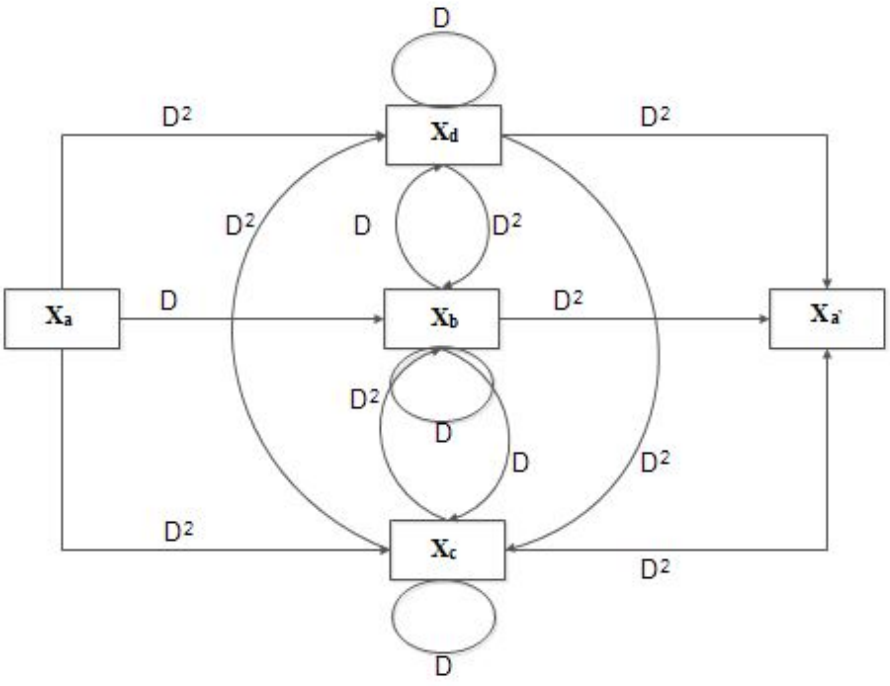


Figure 5.5 signal-flow graph of rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code.

5.3 Decoding of q-ary Convolutional Codes

The decoding of q-ary convolutional codes can be achieved with any of the popular trellis decoding methods, but in this thesis the Max Log-MAP decoder is chosen due to its good tradeoff between performance and complexity. The calculation of the trace forward, trace backwards and branch metric parameters are the same as described in chapter 3, but the overall complexity of the Max Log-MAP algorithm is increased due to the increased number of branches in the trellis. This is illustrated in Fig. 5.6.

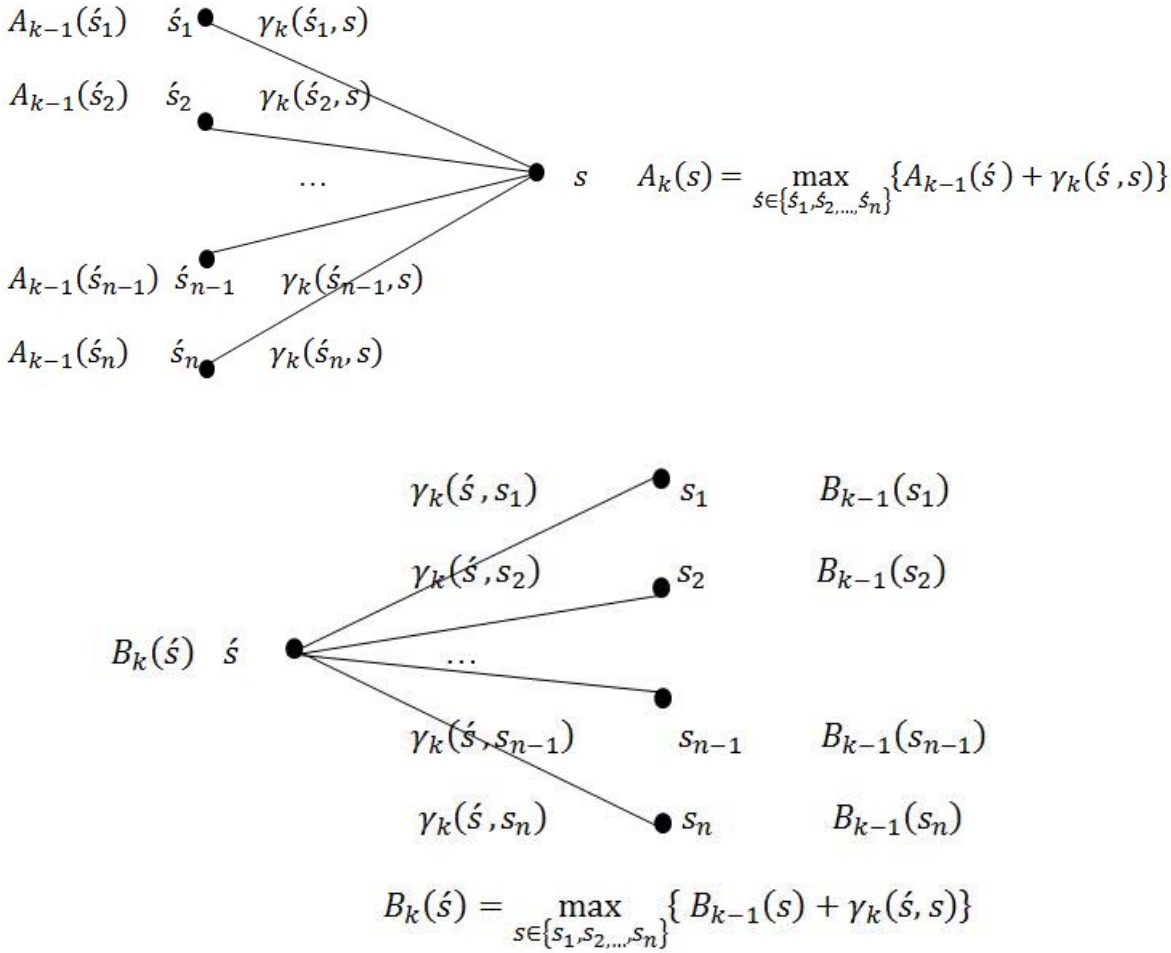


Figure 5.6 The Illustration of trace forward metric *A* and backward metric *B* of MAX-log MAP decoder.

The general equations for the trace and branch metrics for the log-MAP decoder can be written as:

$$A_k(s) = \max_{\acute{s}}\{A_{k-1}(\acute{s}) + \gamma_k(\acute{s}, s)\}, \tag{5.1}$$

$$B_k(\acute{s}) = \max_{s}\{B_k(s) + \gamma_k(\acute{s}, s)\}, \tag{5.2}$$

$$\gamma_k(\acute{s}, s) = -\frac{\|y_k - x\|^2}{2\sigma^2}. \tag{5.3}$$

where $\|y_k - x\|^2$ is the squared euclidean distance between the received symbol y_K and the location of each constellation point x.

When PNC is employed at the relay, then (5.3) must be modified since the resultant constellation diagram due to the summing of the two incoming signals no longer has a unique mapping to a specific finite field symbol. For example, in the 9-point constellation diagram in Figure 3.8 there are four constellation points that map to the symbol 0. Therefore, the conditional probability $P(y_K|c_K = z)$ is given as

$$P(y_K|c_K = z) = \sum_{x \in X_z} P(y_K|x), \tag{5.4}$$

where X_z is the set of constellation points corresponding to the symbol z. So,

$$X_0 \in \{2\sqrt{E} + 2j\sqrt{E},\ 2\sqrt{E} - 2j\sqrt{E},\ -2\sqrt{E} + 2j\sqrt{E},\ -2\sqrt{E} - 2j\sqrt{E}\},$$

$$X_1 \in \{2j\sqrt{E}, -2j\sqrt{E}\},$$

$$X_\beta \in \{2\sqrt{E}, -2\sqrt{E}\},$$

$$X_{\beta^2} \in \{0\}.$$

For the max log-MAP algorithm, we take the natural logarithm of (5.4) to obtain

$$\ln\big(P(y_K|c_K = z)\big) = \ln\left(\sum_{x \in X_z} P(y_K|x)\right),$$

$$\approx \max_{x \in X_z} \left\{ \frac{\|y_k - x\|^2}{2\sigma^2} \right\} \tag{5.5}$$

Hence, $\gamma(\acute{s}, s)$ is expressed as

$$\gamma(\acute{s}, s) = \max_{x \in X_z} \left\{ \frac{\|y_k - x\|^2}{2\sigma^2} \right\} \tag{5.6}$$

Finally, once $A_K(s)$ and $B_K(\acute{s})$ have been determined, the LLR of the decoded message symbols $L^{(z)}(x|y)$ is defined as:

$$L^{(z)}(x_K | y_K) = \ln \left(\frac{\sum_{\acute{s}-s \in S_z} A_{K-1}(\acute{s}) + \gamma_K(\acute{s}, s) + B_K(s)}{\sum_{\acute{s}-s \in S_0} A_{K-1}(\acute{s}) + \gamma_K(\acute{s}, s) + B_K(s)} \right), \tag{5.7}$$

where $L^{(z)}(x_K|y_K)$ is the LLR value corresponding to the finite field element z, S_z is the set of state transitions, s' – s, in the state table where the message symbol is non-zero and S_0 is the set of state transitions where the message symbol is 0. Therefore, $L^{(0)}(x_K|y_K) = 0$ since the numerator and denominator in (5.5) will be the same and it is therefore not used in the decoding of non-binary convolutional codes.

The message symbols can be obtained by making a hard decision on $L^{(z)}(x_K|y_K)$. However, this is not as straightforward as for the binary case because we now have q LLRs for each message symbol. Therefore, the hard decision is made as follows:

- If all $L^{(z)}(x_K|y_K) < 0$, for $z = 1, \beta, \beta^2, \ldots, \beta^{q-2}$, then the decoded message symbol is $m_K = 0$.
- Else the decoded message symbol m_K is equal to the index z of the maximum value of $L^{(z)}(x_K|y_K)$, for $z = 1, \beta, \beta^2, \ldots, \beta^{q-2}$

5.4 Theoretical Performance of q-ary Convolutional Codes

In chapter 3, the theoretical performance of binary convolutional codes was explained, which required knowledge of the transfer function of the code derived from the signal flow graph. The theoretical performance of non-binary convolutional codes can be derived using the same method, but we now find that the signal flow graph is more complex and the state equations will have more terms due to the increased size of the message alphabet.

Recall the signal-flow graph in Figure 5.5 for the rate ½ $\beta\beta^2/1$ convolutional code defined in GF(4).

Let X_a= state 0, X_b= state 1, X_c= state β and X_d= state β^2. Thus, we can write the four state equations:

$$X_b = D^2X_a + DX_b + DX_c + D^2X_d, \tag{5.8}$$

$$X_c = DX_a + D^2X_b + DX_c + D^2X_d, \tag{5.9}$$

$$X_d = D^2X_a + D^2X_b + DX_c + D^2X_d, \tag{5.10}$$

$$X_{a'} = D^2X_b + D^2X_c + D^2X_d, \tag{5.11}$$

Where D and D^2 are referring the hamming distance which compares the symbol difference instead of bit difference.

By definition the transfer function T(D) is defined as:

$$\mathrm{T(D)} = \frac{X_{a'}}{X_a} = \frac{D^2X_b + D^2X_c + D^2X_d}{X_a} \tag{5.12}$$

$$= 2D^4 + D^3 + 2D^2(2D^2 + D)\frac{X_b}{X_a} + 3D^3\frac{X_c}{X_a}.$$

From (5.8)-(5.11), it can be concluded that:

$$X_b = \frac{D^2 X_a + \mathrm{D} X_c}{1 - D - D^2}, \tag{5.13}$$

and

$$X_c = \frac{\mathrm{D} X_a + 2D^2 X_b}{1 - D}. \tag{5.14}$$

So, by substituting (5.13) to (5.14), we can have the relationship of $\frac{X_b}{X_a}$:

$$\frac{X_b}{X_a} = \frac{2D^2 - D^3}{1 - 2D - D^3}. \tag{5.15}$$

Also, by following the same procedure, the relationship of $\frac{X_c}{X_a}$ can be derived as:

$$\frac{X_c}{X_a} = \frac{D - D^2 + D^3}{1 - 2D - D^3}. \tag{5.16}$$

Thus, by substituting (5.13) and (5.14) to (5.10), we can have:

$$\mathrm{T(D)} = \frac{D^3 + 3D^4 - 3D^5 + 8D^6 - 6D^7}{1 - 2D - D^3}. \tag{5.17}$$

Then the transfer function T(D) can be derived by performing the long-division, resulting in:

$$\mathrm{T(D)} = D^3 + 5D^4 + 7D^5 \ldots \tag{5.18}$$

which means the rate ½ $\beta\beta^2$/1 4-ary convolutional code has one path of $d_{min} = 3$, and five paths with $d = 4$, and so on. Thus, recall (5.1), the bound of bit error probability of a rate $\frac{1}{2}$ $\beta\beta^2$/1 4-ary convolutional code can be calculated as

$$P \leq \frac{2^{k-1}}{2^k - 1} \frac{1}{K} \sum_{w} \sum_{d_{free}}^{N} T_d P_{channel}. \tag{5.19}$$

where the term $\frac{2^{k-1}}{2^k-1}$ converts symbol error rate (SER) to BER [73], and $\mathrm{k} = \log_2 q$ of GF(q).

5.5 PNC combined with q-ary Convolutional Codes

The system models of q-ary convolutional codes applied to the TWRC with PNC are shown in Figure 5.7 and Figure 5.8, respectively.

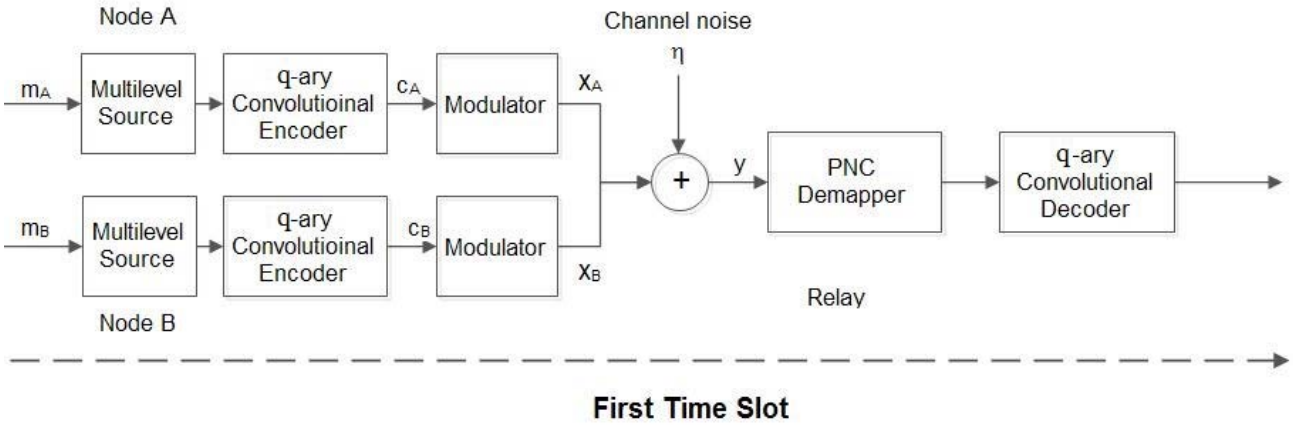


Figure 5.7 System model of q-ary convolutional code decoding at the relay on PNC.

In Fig. 5.7, the q-ary convolutional decoder is decoding the received signal y at the relay, where $y = X_A + X_B + \eta$. Recalling (3.13) and substituting into (5.19), the upper bound on the bit error probability P_{ncr} in this case is:

$$P_{ncr} \leq \frac{2^{k-1}}{2^k - 1} \frac{1}{K} \sum_{w} \sum_{d_{free}}^{N} T_d P_{qr}. \tag{5.20}$$

The bound on the theoretical BER of decoding at the destination nodes is derived as

$$P_{ncn} \leq (1 - P_{channel}) P_{ncr} + (1 - P_{ncr}) P_{channel} - 2 P_{channel} P_{ncr}. \tag{5.21}$$

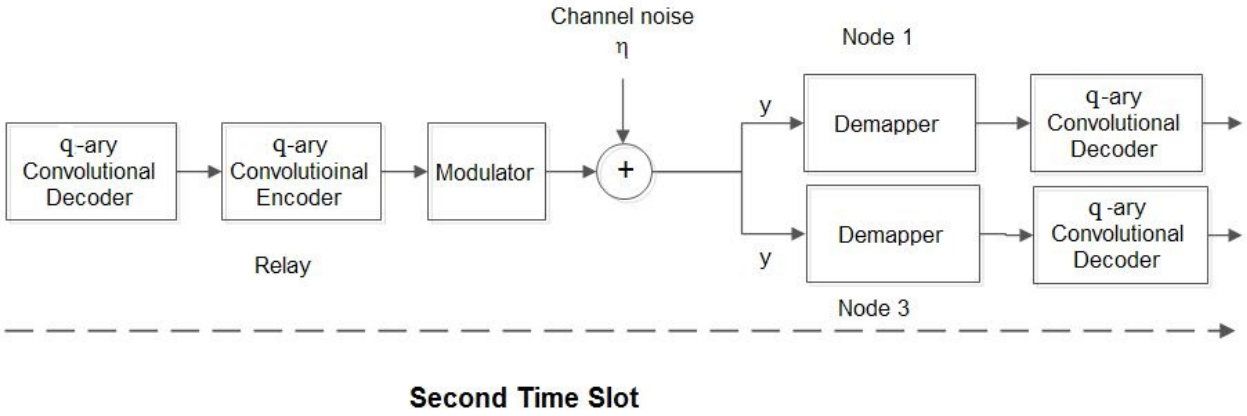


Figure 5.8 System model of q-ary convolutional codes decoding at the nodes on PNC over impulsive channel.

The comparison of simulated BER to theoretical BER of rate $\frac{1}{2}$ $\beta\beta^2$ /1 4-ary convolutional code on PNC over AWGN channel is presented in Figure 5.9.

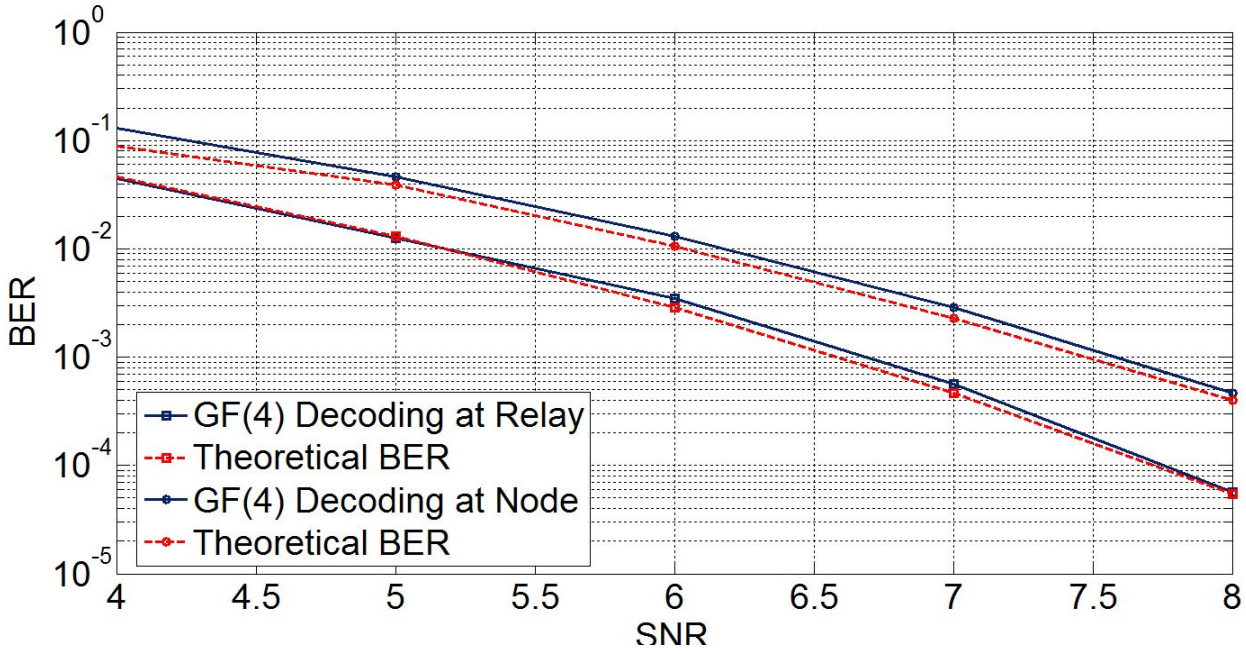


Figure 5.9. The comparison of simulation BER to theoretical BER of rate $\frac{1}{2}$ $\beta\beta^2$/1 4-ary convolutional code on PNC over AWGN channel.

As shown in Figure 5.9, the simulated BERs match the bounds closely, which proves the derivation of the bounds is correct. Not surprisingly, the BER performance of decoding at the relay is approximately 1dB better than decoding at the node.

So far the analysis of PNC combined with the binary convolutional code is on the AWGN channel, but now the performance is evaluated on impulsive noise channel using the Gaussian mixture model. Recalling (3.17) and submitting in (5.19), the bound of rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code on PNC over impulsive channel p_{ncr}^{GMM} can be expressed as:

$$p_{ncr}^{GMM} \leq \frac{2^{k-1}}{2^k - 1} \sum_{w} \sum_{d_{free}}^{N} T_d p_{qr}^{GMM} . \tag{5.22}$$

Also the bound of decoding at the node p_{ncn}^{GMM} in this case is:

$$p_{ncr}^{GMM} = (1 - P_{GMM}) p_{ncr}^{GMM} + (1 - p_{ncr}^{GMM}) P_{GMM} - 2 P_{GMM} p_{ncr}^{GMM}. \tag{5.23}$$

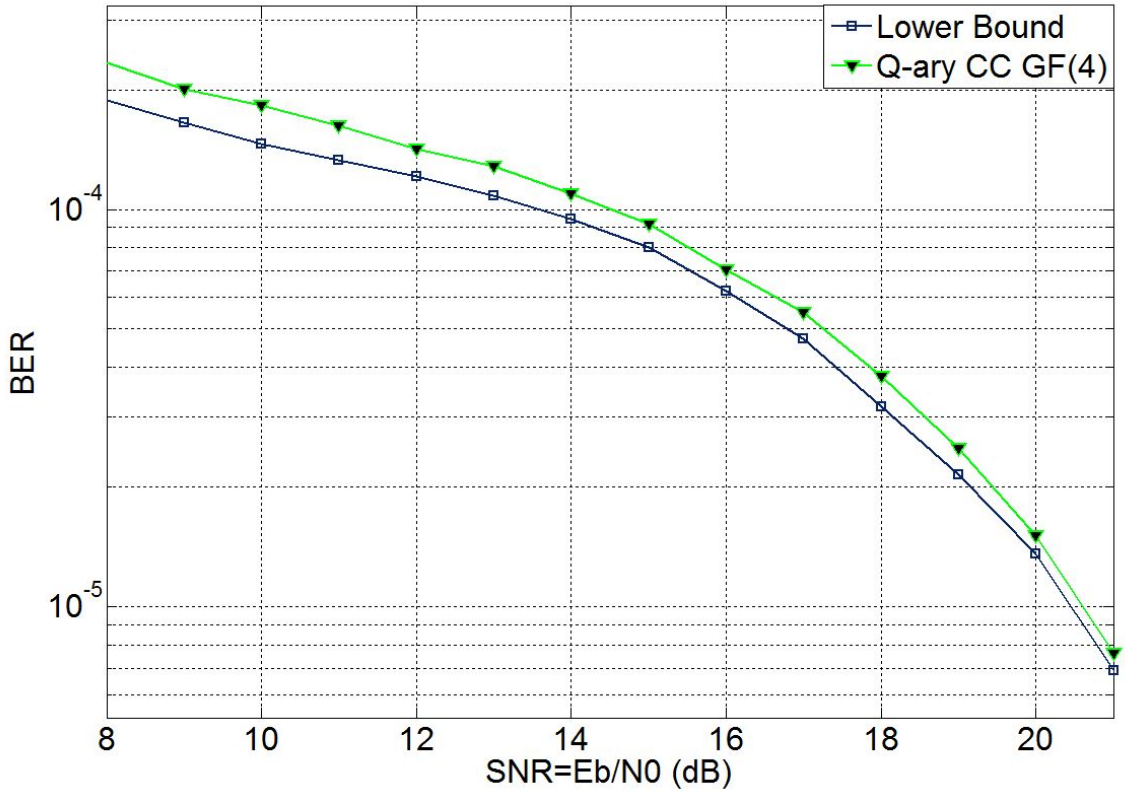


Figure 5.10. The comparison of simulated BER to theoretical BER of rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code on PNC over impulsive channel decoding at the relay, mixture constant $\alpha = 0.1$.

The comparison of simulated BER with the theoretical BER of PNC combined with the rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code over an impulsive noise channel with $\alpha = 0.1$ are shown in Figure 5.10 and Figure 5.11.

From Figure 5.10 and Figure 5.11, it can be seen that the simulated BER performance is only slightly degraded at a BER of 10^{-5}. Hence, the simulated BER converges to the theoretical BER, which verifies the results. However, with ten percent of impulsive noise present in the channel, the BER degradation of the rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code is more significant, which has a difference of approximately 8dB compared to the performance on the AWGN channel.

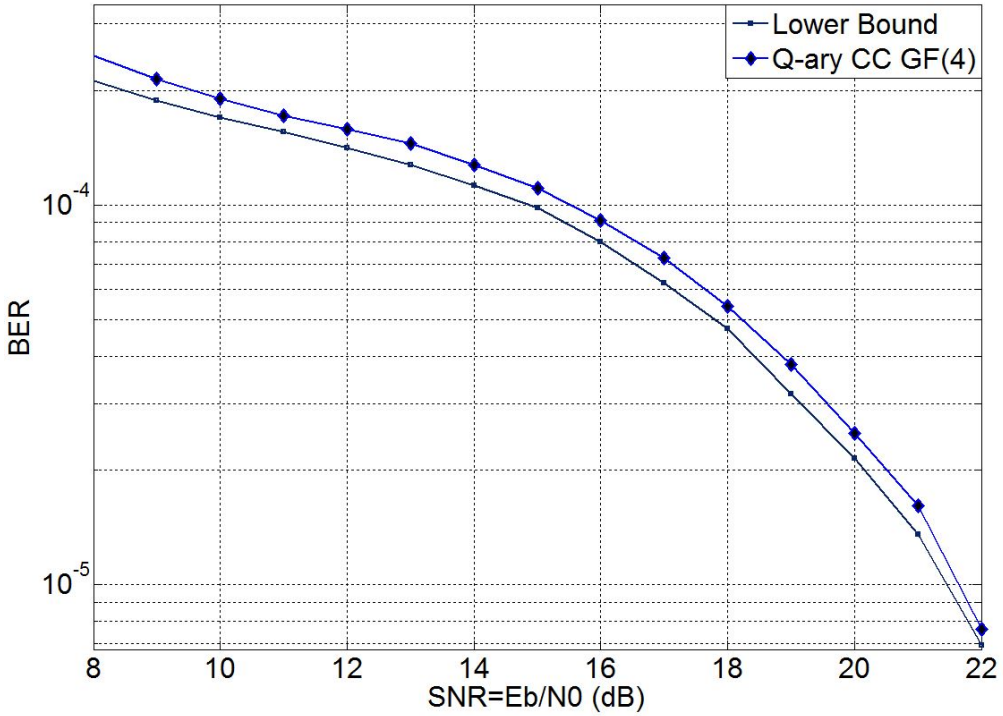


Figure 5.11. The comparison of simulated BER to theoretical BER of rate $\frac{1}{2}$ $\beta\beta^2/1$ 4-ary convolutional code on PNC over impulsive channel decoding at the node, mixture constant α = 0.1.

5.6 Simulated Performance of PNC combined with Non-Binary Convolutional Codes on Impulsive Noise Channels

The following section presents several results for convolutional codes defined in GF(2), GF(4) and GF(16) on impulsive noise channels with difference mixtures of impulses. There are very few papers containing tables of good non-binary codes defined in different fields, so the convolutional code defined in GF(16) has been selected from [59], which catalogues a number of good non-binary convolutional codes. However, all codes in [60] are non-systematic and comprise only feedforward coefficients. Therefore, for a fair comparison the other codes defined in GF(2) and GF(4) are also chosen to be non-systematic. Furthermore, the number of states is fixed at 256 for all the codes and the message length in bits is fixed at 1000 bits. For binary convolutional codes we choose the $(561,753)_8$ code since it has the maximum minimum Hamming distance for a rate ½ code with a constraint length of 9. The mixture parameter is set to $\alpha = 0, 0.01$ and 0.1, corresponding to an AWGN channel, moderately impulsive channel and heavily impulsive channel respectively. The parameters of the q-ary convolutional codes are listed in Table 5.2.

	GF(2)	GF(4)	GF(16)
Constraint Length	9	5	3
d_{min}	12 13 14	9 10 11	6 7 8
$N_{d_{free}}$	11 50 286	12 39 303	15 120 900
Generator Polynomials	$(561,753)_8$	$1\ 1\ 1\ \beta^2\ \beta$ $1\ \beta\ 1\ \beta\ \beta^2$	$1\ 1\ \beta^4$ $1\ \beta\ \beta^4$

Table 5.2. Parameters of rate $\frac{1}{2}$ q-ary convolutional codes, the primitive polynomials are $x^2 + x + 1$ and $x^4 + x + 1$ for GF(4) and GF(16) respectively.

The performance comparison of PNC combined with the different convolutional codes at the relay over an impulsive channel with different mixtures is shown in Figure 5.12. At the relay, any significant positive or negative impulses are clipped so that their energy is no greater than the transmitted symbol energy *E*. From the figure it can be seen that the performance of coded PNC is severely affected on additive impulsive noise channels compared to the AWGN channel. When $\alpha = 0$ all codes converge at a BER of 10^{-5}, but the q-ary convolutional codes perform slightly better at low SNR. The binary convolutional code shows a higher error floor compared with the q-ary convolutional codes when $\alpha > 0$. Surprisingly, the convolutional code over GF(4) performs the best with a lower error floor than the convolutional code over GF(16). The binary convolutional code performs the worst for all values of α.

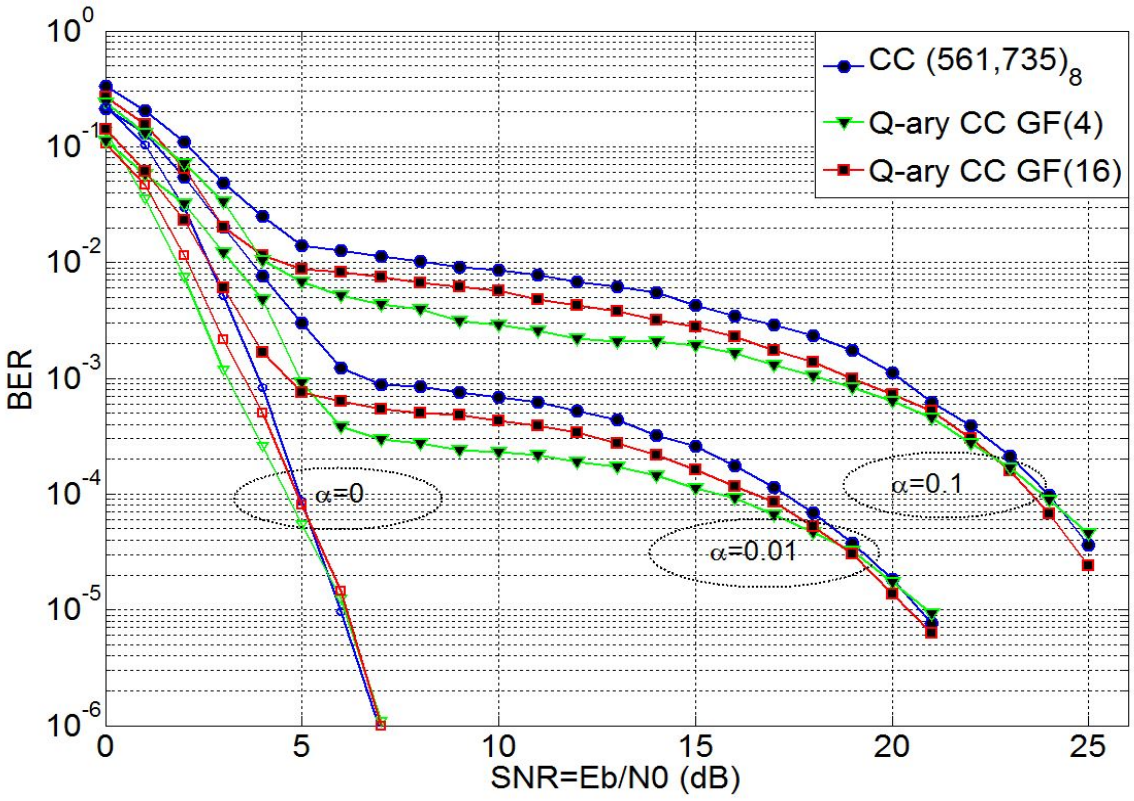


Figure 5.12 Simulation results for PNC combined with binary and non-binary convolutional codes on impulsive noise channels with $\alpha = 0, 0.01$ and 0.1.

We observe in fig. 5.12 that the performance of the code over GF(16) does not outperform the code over GF(4), which at first seems unexpected. However, it must be noted that the number of states in the trellis of all codes is fixed at 256. This means the code over GF(16) having a smaller constraint length, which results in a smaller minimum Hamming distance and reduces the error-correcting capabilities of the code. For example, the code over GF(16) has a minimum Hamming distance of 6, but the code over GF(4) has a minimum Hamming distance of 9.

The error floor discussion is grouped by different values of α : First, when the channel is slightly impulsive with $\alpha = 0.01$, the performances of all codes are significantly degraded by the impulsive noise, with a coding loss of around 15dB for all codes at a BER of 10^{-5}. However, the BER performance of all codes beings to converge as SNR exceeds 20dB. Secondly, when the noise in channel becomes more impulsive at $\alpha = 0.1$, the BER performance of all codes converge at a BER of 10^{-4} when the SNR exceeds 24dB. In this case, the performance is further degraded by the increased impulsiveness, with a coding loss of around 18dB at a BER of 10^{-4}.

5.7 Design of q-ary Convolutional Codes for use with PNC on Impulsive Noise Channels

One of the biggest advantages of a q-ary convolutional code is that the codewords are constructed with symbols instead of single information bits, for which the decoder can correct several bits at a time. For example, each symbol of a K=5 convolutional code defined in GF(4) contains two bits so with a free distance of 9 this code can correct up to 4 symbol errors. Therefore, if the bit errors are close together, i.e. a burst error extending over 4 symbols, then this code can correct up to 8 bit errors. However, if the bit errors are spread then we can have the situation where each symbol only contains a single bit error, meaning that the code can only correct up to 4 bit errors in this case. This is why the performance of non-binary codes tends to converge with binary codes on the AWGN channel at high SNRs, but we find that on impulsive noise channels the errors are closer

together and the non-binary code has a performance advantage over the binary code, particularly at the SNR regions where the error floors occur.

Recall (5.19) and rewrite as:

$$P_{ncr} \leq \frac{2^{k-1}}{2^k - 1} \frac{1}{K} \sum_{w} \sum_{\mathrm{N}_{d_{free}}}^{N} T_{d_{free}} P_{qr}. \tag{5.24}$$

The effects of various $\mathrm{N}_{d_{free}}$ and d_{free} for convolutional codes in different finite fields are now shown in Figure 5.13 and Figure 5.14. The q-ary convolutional codes with small $\mathrm{N}_{d_{free}}$ have better performance in the error floor region. The comparisons are made by choosing $\mathrm{N}_{d_{free}} = 5, 10$ and 15 with a fixed d_{free}, and $d_{free} = 8, 10$ and 12 with a fixed $\mathrm{N}_{d_{free}}$. In Figure 5.13, it can be seen that when $\alpha = 0$, increasing values of $\mathrm{N}_{d_{free}}$ only slightly degrade the performance of the code but it generally has a small effect. However, when $\alpha > 0$ codes with smaller $\mathrm{N}_{d_{free}}$ have a significantly lower error floor. For example, when $\alpha = 0.01$, there is a coding loss of up to 8dB when comparing the code with $\mathrm{N}_{d_{free}}$ = 5 and $\mathrm{N}_{d_{free}}$ = 15 at a BER of 3×10^{-4}.

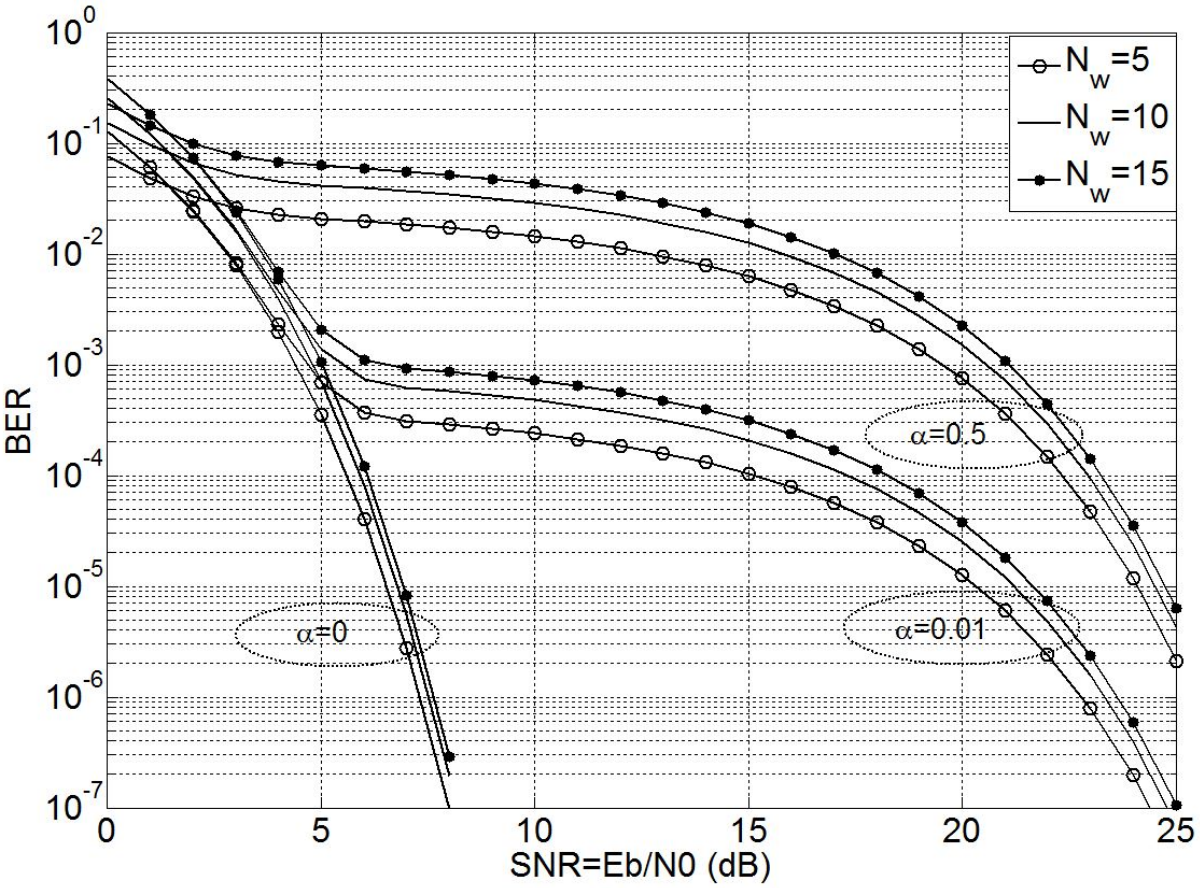


Figure 5.13. Comparison of theoretical BER with different N_w with fixed d_{free} of q-ary convolutional code on PNC over Impulsive channel.

Similarly, coding losses of up to 10dB occur when comparing the same values of $N_{d_{free}}$ for $\alpha = 0.5$ at a BER of 2×10^{-4}. The effect of different d_{free} and fixed $N_{d_{free}}$ are shown in Figure 5.13. It can be seen that when $\alpha = 0$, coding gains of 1dB are achieved as d_{free} is increased from 8 to 10 to 12 at a BER of 10^{-6}. When the channel is impulsive, d_{free} has a negligible effect on the error floor. However, at high SNR coding gains of 1dB are observed as before for both $\alpha = 0.01$ and $\alpha = 0.5$.

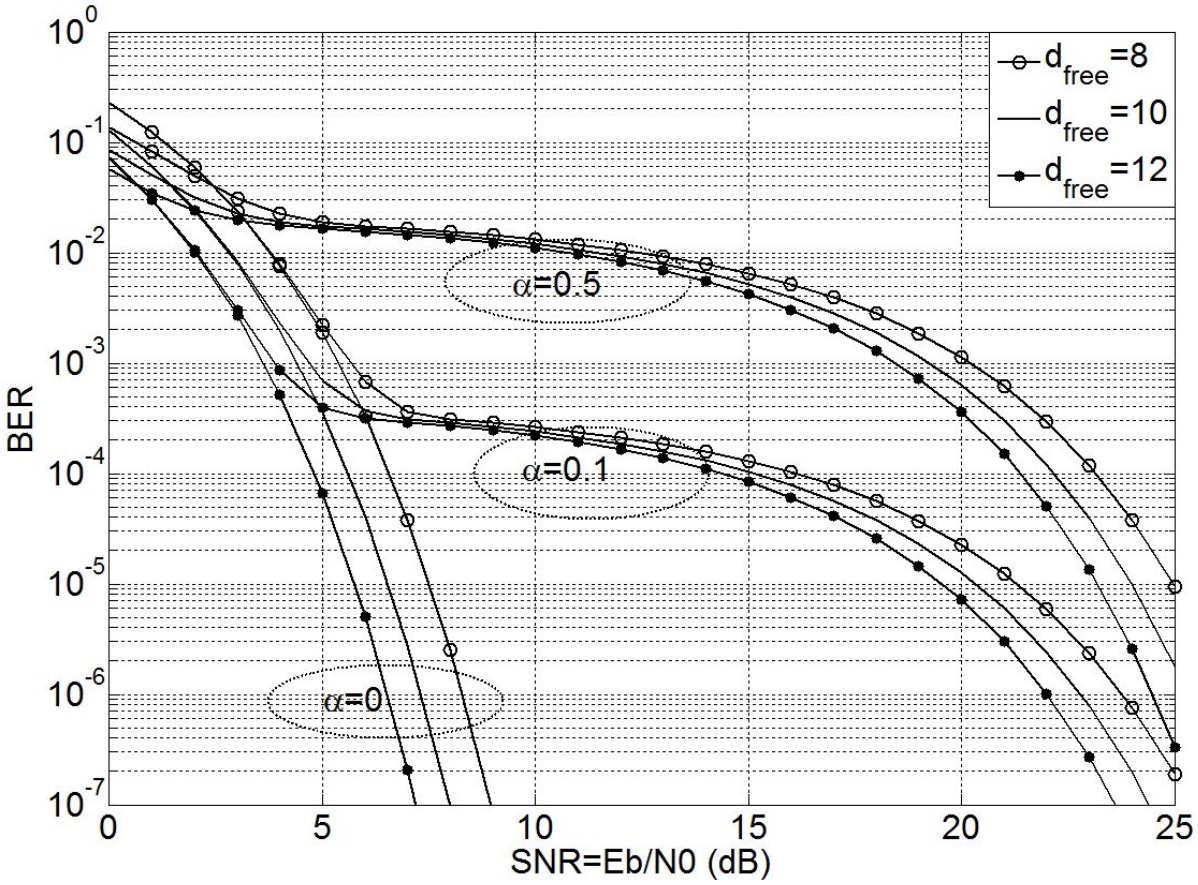


Figure 5.14. Comparison of theoretical BER with different d_{free} with fixed $N_{d_{free}}$ of q-ary convolutional code on PNC over Impulsive channel.

In order to make a comparison to the binary convolutional codes to show the advantage of non-binary convolutional codes via theoretical bounds, we compare the theoretical performances of convolutional codes to the non-binary convolutional under the same conditions, i.e. $N_{d_{free}} = 5$ and $d_{free} = 12$, shows in Fig. 5.15:

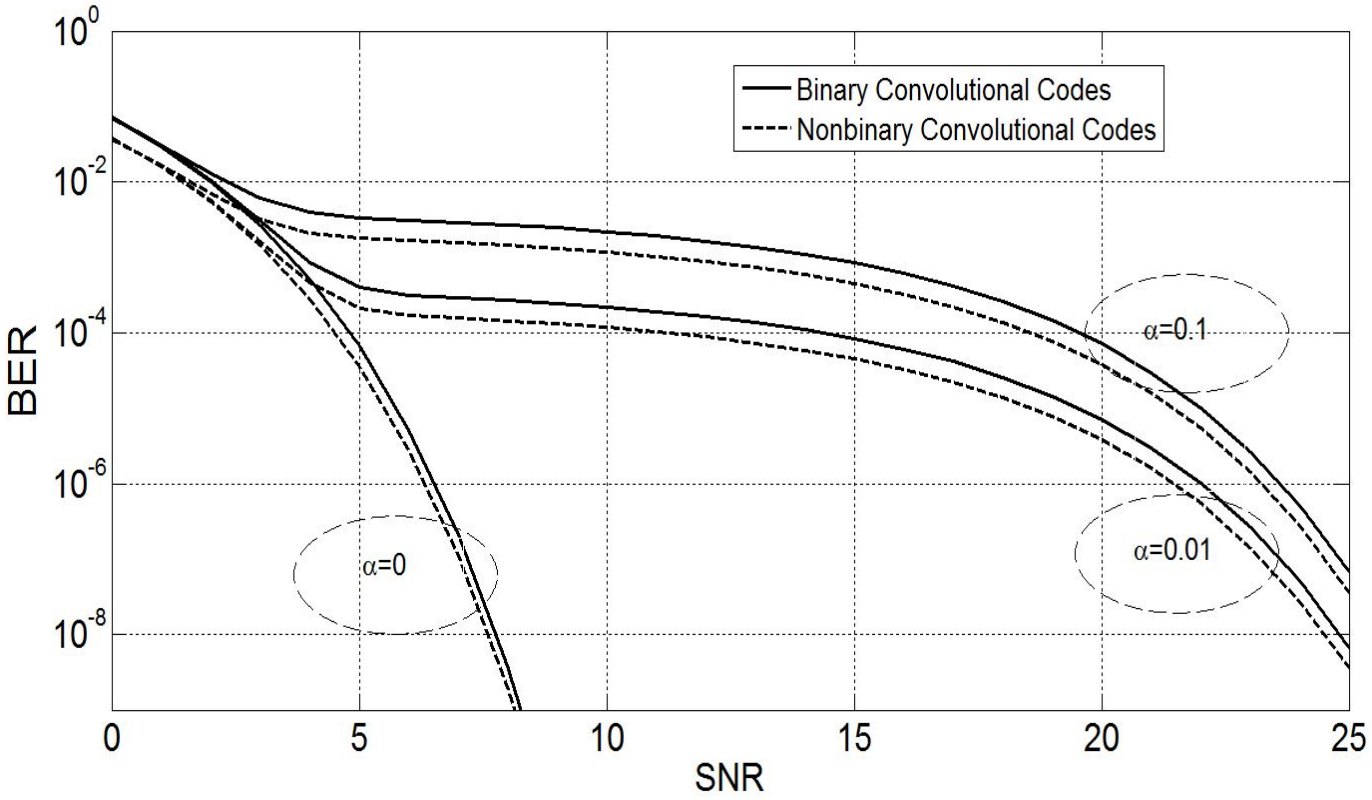


Figure 5.15 Theoretical BER comparison of GF(16) non-binary convolutional codes to binary convolutional codes on PNC over impulsive noise channel with $N_{d_{free}} = 5$ and $d_{free} = 12$.

From Fig 5.15 it can be seen that the GF(16) non-binary convolutional codes outperform the binary convolutional codes under all circumstances with the same $N_{d_{free}}$ and d_{free}. The performance of non-binary convolutional codes on AWGN channel is only better at lower SNR, and soon converges to the binary convolutional code at $7dB$. However, the advantage of non-binary convolutional codes is significant on impulsive channels. Clearly, with a lower error floor level the non-binary convolutional codes have a $3dB$ coding gain at a BER of 10^{-4} when $\alpha = 0.01$, and a $2dB$ coding gain at BER of 10^{-4} when $\alpha = 0.1$. Although both codes are converging at higher SNR, the coding gain still exists even at a high SNR of $25dB$. Thus, the non-binary convolutional codes have shown their advantage over binary convolutional codes and have a great potential for application on impulsive noise channels.

5.8 Conclusions

In this chapter, the performance of PNC combined with binary and non-binary convolutional code on impulsive noise channels has been presented. The chapter began by explaining the fundamental theory of non-binary convolutional codes, which included encoder structure, the log-MAP decoder with non-binary LLR values and a upper bound on the theoretical BER performance of these codes at the relay of a TWRC employing PNC and at the destination nodes. The theoretical BERs were compared with simulation results for both binary and non-binary codes and it was observed that on the AWGN channel there was no performance advantage for non-binary codes over binary codes at high SNR. However, on impulsive noise channels it could be clearly seen that the non-binary codes had a lower error floor region, achieving significant coding gains, before converging with the binary codes at high SNRs.

Finally, the design of convolutional codes for use in a TWRC with PNC was presented by investigating the effect of the parameters d_{free} and $N_{d_{free}}$. Three convolutional codes defined in GF(2), GF(4) and GF(16) were investigated by plotting their theoretical BER performance by varying $N_{d_{free}}$ and keeping d_{free} fixed and also by varying d_{free} and keeping $N_{d_{free}}$ fixed. It was observed that on the AWGN channel, the performance of PNC with binary and non-binary convolutional codes is dominated by d_{free}, whereas increasing $N_{d_{free}}$ only slightly degraded performance. However, on impulsive noise channels it was observed that decreasing the value of $N_{d_{free}}$ had a more significant effect on performance by lowering the error floor, whereas increasing d_{free} had a negligible effect on the error floor, for all levels of impulsiveness. As before, d_{free} only effected performance at high SNRs. From these observations it is clear that choosing convolutional codes with a higher value of $N_{d_{free}}$ will results in a higher undesirable error floor and minimizing $N_{d_{free}}$ is the most important design criterion when designing convolutional codes for use in a PNC system.

Chapter 6

Physical Layer Network Coding combined with Non-Binary Turbo Codes

6.1 Introduction

In the previous chapter, the performance and design of non-binary convolutional codes for use with a TWRC employing PNC were investigated and it was observed that non-binary codes exhibit lower error floors than binary codes on impulsive noise channels. Following on from these promising results, this chapter will investigate PNC combined with non-binary turbo codes with the aim of achieving much greater performances on impusive noise channels. There is very little research on non-binary turbo codes in the literature and apparently no research on applications of non-binary turbo codes. Therefore, this chapter presents very original and interesting results that show for the first time how non-binary turbo codes behave at the relay of a TWRC with PNC and also how their performance is affected by impulsive noise. The non-binary turbo encoder will be introduced and modifications to the log-MAP decoding algorithm from chapter 5 will be explained in order to realise the non-binary turbo decoder. A detailed worked example of the non-binary turbo decoder is presented showing how the received symbols at the relay are decoded after PNC demapping. Simulation results of the performance of binary and non-binary turbo codes at the relay on AWGN and impulsive noise channels. Finally, theoretical upper bounds on the BER performance of non-binary turbo codes defined in GF(4) and GF(16) are derived to predict the error floor of these codes at the relay on AWGN and impulsive noise channels and are compared with the error floor upper bound of binary turbo codes.

6.2 Non-binary Turbo Encoder

A non-binary turbo encoder is almost identical to a binary turbo encoder, with the difference that it comprises two non-binary recursive systematic convolutional (RSC) codes in parallel separated by an interleaver. The encoder of a non-binary turbo encoder is shown in Figure 6.1.

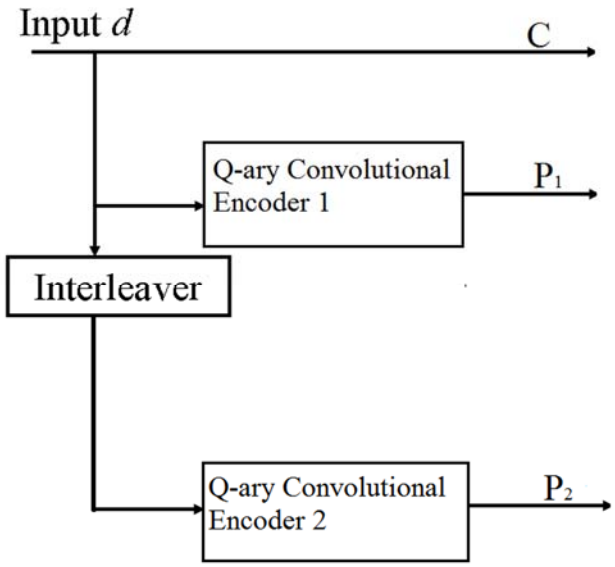


Figure 6.1 Encoder structure of non-binary turbo code.

The message symbols input d and the encoder output $C,\ P_1, P_2$ are now defined over finite fields, $GF(q)$. As before, this arrangement results in a rate 1/3 turbo code, but puncturing can be applied to P_1 and P_2 as explained in chapter 4. In this chapter, the convolutional codes used are defined in GF(4) and GF(16) and have feedforward and feedback coefficients of $\beta\beta^2/1$ and $\beta^7\beta^4/\beta^6$ respectively. The $\beta\beta^2/1$ RSC code has 4 states and the $\beta^7\beta^4/\beta^6$ RSC code has 16 states.

6.3 Non-binary Turbo Decoder

In Fig 6.2, the soft values entering the turbo decoder are LLRs for the non-binary symbols of the message and parity-check symbols of the transmitted codeword. As mentioned in chapter 5, there are $q-1$ LLR values corresponding to the reliability of a

received symbol corresponding to $0,1,\beta,\beta^2,\ldots,\beta^{q-2}$ compared to the zero symbol. In non-binary turbo decoders, we express the output LLR in a multi-dimensional way:

$$L^{(z)}(c_K|y) = \ln\left(\frac{P(c_K = z|y)}{P(c_K = 0|y)}\right) \tag{6.1}$$

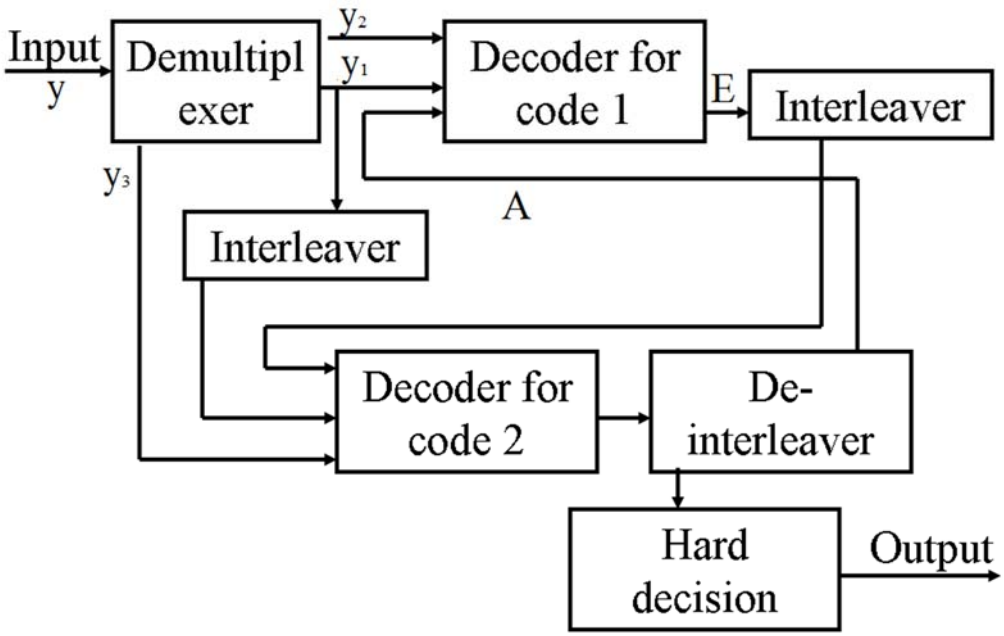


Figure 6.2 Decoder structure of non-binary turbo decoder.

where $z = GF(q)$. Thus, for GF(4), (6.1) comprises the following LLRs:

$$\begin{Bmatrix} L^{(0)}(c_K|y) = \ln\left(\frac{P(c_K = 0|y)}{P(c_K = 0|y)}\right) = 0 \\ L^{(1)}(c_K|y) = \ln\left(\frac{P(c_K = 1|y)}{P(c_K = 0|y)}\right) \\ L^{(\beta)}(c_K|y) = \ln\left(\frac{P(c_K = \beta|y)}{P(c_K = 0|y)}\right) \\ L^{(\beta^2)}(c_K|y) = \ln\left(\frac{P(c_K = \beta^2|y)}{P(c_K = 0|y)}\right) \end{Bmatrix}. \tag{6.2}$$

6.4 Non-binary Turbo Codes on PNC

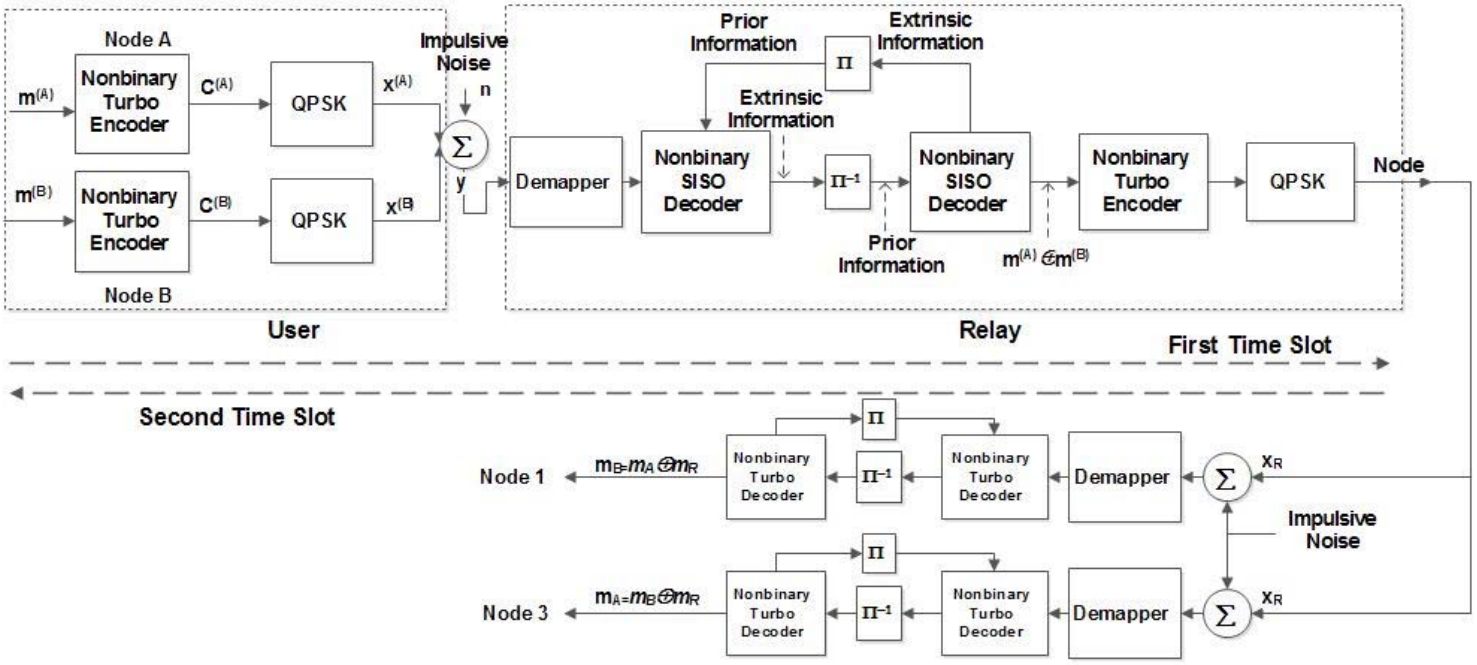


Figure 6.3 System model of non-binary turbo codes on PNC over impulsive channel.

The system model of the non-binary turbo code combined with PNC is shown in Figure 6.3 and we are focused on the performance of non-binary turbo code at the relay. Let $m_A, m_B \in [0,1,\beta,\beta^2]$ denote the information from both source nodes and $c_A, c_B \in [0,1,\beta,\beta^2]$ denote their codewords respectively. At the relay, the incoming QPSK symbols are added at the relay and result in a nine-point constellation diagram. As with the binary turbo decoder in chapter 4, to decode the added signal at the relay, we are using max-log-MAP decoders for the component decoders. Thus, the trace metrics A, B and branch metric γ can be expressed as:

$$A_K(s) = \max_{\acute{s}}\{A_{K-1}(\acute{s}) + \gamma_K(\acute{s}, s)\}, \tag{6.3}$$

$$B_K(\acute{s}) = \max_{s}\{B_K(s) + \gamma_K(\acute{s}, s)\}, \tag{6.4}$$

$$\gamma_K(\acute{s}, s) = L_a^{(z)}(k) + \max_{x \in X_i} \left\{ -\frac{\|y_K - x\|^2}{2\sigma^2} \right\} \tag{6.5}$$

A detailed worked example of one complete decoding iteration is now presented to explain the decoding process of non-binary turbo codes. The message length is set to 10 and the interleaver mapping is defined as:

10	4	5	9	3	7	8	6	1	2

The messages from node 1 and node 3 are:

$\mathrm{m}_A =$

$\boldsymbol{k}$	0	1	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	β^2	1	1	β^2	β	β	β^2	β^2	β^2	β^2

and

$\mathrm{m}_B =$

$\boldsymbol{k}$	0	1	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	β	β	0	β	β	β^2	β	β^2	β^2	1

At the relay, the 30-symbol received vector, corresponding to the sum of the transmitted QPSK symbols plus noise, is split into three 10-symbol vectors corresponding to the summed message y_1 , parity-check symbols from encoder 1 y_2 and parity-check symbols from decoder 2 y_3.

$\mathrm{y}_1 =$

k	0	1	2	3	4	5	6	7	8	9
	-0.21 - 1.7j	0.56+ 0.5j	0.2 + 1.86j	0.27 - 1.74j	2.8 – 2.9j	0.3 + 0.18j	-2.75 - 1.7j	-2.57 + 0.4j	-2.57 + 1.2j	3.1 + 1.62j

$y_2 =$

k	0	1	2	3	4	5	6	7	8	9
	-2.6+ 0.39j	-2.6 + 1.1j	3.1 + 1.6j	-0.2 - 1.7j	0.57 + 0.2j	0.18 + 1.8j	0.27 - 1.74j	1.9 + 0.66j	0.3 + 0.18j	-2.76 - 1.7j

$y_3 =$

k	0	1	2	3	4	5	6	7	8	9
	1.9+ 0.67j	0.3 + 0.19j	-2.75 - 1.7j	-2.56 + 0.4j	-2.57 + 1.1j	3.1 + 1.6j	0.2 + 1.86j	0.57 + 0.2j	0.2 + 1.86j	0.27 - 1.75j

The LLR values corresponding to the summed message m_R, denoted by $L\,y_1$ after PNC demapping are :

$L_{y1} =$

$k/_{L(z)}$	0	1	2	3	4	5	6	7	8	9
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	3.4	3.4	3.4	3.4	-6.7	3.4	3.4	-3.4	-3.4	-3.4
β	-3.4	3.4	-3.4	3.4	6.7	-3.4	3.4	-3.4	-3.4	3.4
β^2	0.0	6.7	0.0	6.7	0.0	0.0	6.7	-6.7	-6.7	0.0

In this example, noise has caused the fourth symbol of y_1 to introduce an error and a hard decision of L_{y1} generates the summed codeword:

$c_R =$

k	0	1	2	3	4	5	6	7	8	9
	1	β^2	1	1	β	1	1	0	0	β

The a priori LLRs that are fed into decoder 1 are initialised to zero, so by applying (6.5), we can calculate γ:

$\gamma =$

$k/(s,s')$	0	1	2	3	4	5	6	7	8	9
$\mathbf{0 \to 0}$	-6.7	-6.7	-3.4	-6.7	-11.7	-3.4	-10.1	-3.4	-6.7	-10.1
$\mathbf{0 \to 1}$	0.0	-6.7	-3.4	-6.7	-15.1	-3.4	-10.1	-3.4	-6.7	-10.1
$\mathbf{0 \to \beta}$	-10.1	-10.1	-13.4	-10.1	-5.0	-13.4	-6.7	-6.7	-3.4	0.0
$\mathbf{0 \to \beta^2}$	-10.1	-3.4	-6.7	-3.4	-15.1	-6.7	0.0	-13.4	-10.1	-6.7
$\mathbf{1 \to 0}$	-10.1	-10.1	-6.7	-10.1	-15.1	-6.7	-6.7	-6.7	-3.4	-6.7
$\mathbf{1 \to 1}$	-3.4	-10.1	-6.7	-10.1	-18.5	-6.7	-6.7	-6.7	-3.4	-6.7
$\mathbf{1 \to \beta}$	-6.7	-6.7	-10.1	-6.7	-1.7	-10.1	-10.1	-3.4	-6.7	-3.4
$\mathbf{1 \to \beta^2}$	-6.7	0.0	-3.4	0.0	-11.7	-3.4	-3.4	-10.1	-13.4	-10.1
$\mathbf{\beta \to 0}$	-3.4	-10.1	-6.7	-10.1	-8.4	-6.7	-13.4	0.0	-3.4	-6.7
$\mathbf{\beta \to 1}$	-3.4	-3.4	0.0	-3.4	-18.5	0.0	-6.7	-6.7	-10.1	-13.4
$\mathbf{\beta \to \beta}$	-13.4	-6.7	-10.1	-6.7	-8.4	-10.1	-3.4	-10.1	-6.7	-3.4
$\mathbf{\beta \to \beta^2}$	-6.7	-6.7	-10.1	-6.7	-11.7	-10.1	-3.4	-10.1	-6.7	-3.4
$\mathbf{\beta^2 \to 0}$	-6.7	-13.4	-10.1	-13.4	-11.7	-10.1	-10.1	-3.4	0.0	-3.4
$\mathbf{\beta^2 \to 1}$	-6.7	-6.7	-3.4	-6.7	-21.8	-3.4	-3.4	-10.1	-6.7	-10.1
$\mathbf{\beta^2 \to \beta}$	-10.1	-3.4	-6.7	-3.4	-5.0	-6.7	-6.7	-6.7	-10.1	-6.7

$\boldsymbol{\beta^2 \rightarrow \beta^2}$	-3.4	-3.4	-6.7	-3.4	-8.4	-6.7	-6.7	-6.7	-10.1	-6.7

The two trace metrics A and B are then calculated using (6.3) and (6.4):

$A =$

$\boldsymbol{k/_{\acute{S}}}$	0	1	2	3	4	5	6	7	8	9	10
0	0	-6.7	-10.1	-10.1	-3.4	-15.1	-11.7	-11.7	-15.1	-18.5	-18.5
1	$-\infty$	0.0	-10.1	-10.1	-3.4	-15.1	-11.7	-11.7	-15.1	-18.5	-18.5
$\boldsymbol{\beta}$	$-\infty$	-10.1	0.0	-6.7	-6.7	-8.4	-8.4	-11.7	-11.7	-15.1	-18.5
$\boldsymbol{\beta^2}$	$-\infty$	-10.1	-6.7	0.0	-10.1	-5.0	-8.4	-11.7	-15.1	-15.1	-18.5

$B =$

$\boldsymbol{k/_{\acute{S}}}$	0	1	2	3	4	5	6	7	8	9	10
0	-18.5	-28.5	-21.8	-21.8	-15.1	-13.4	-10.1	-10.1	-6.7	-10.1	0
1	-28.5	-18.5	-25.2	-18.5	-18.5	-16.8	-10.1	-10.1	-10.1	-6.7	$-\infty$
$\boldsymbol{\beta}$	-25.2	-28.5	-18.5	-21.8	-18.5	-10.1	-13.4	-6.7	-6.7	-3.4	$-\infty$
$\boldsymbol{\beta^2}$	-28.5	-25.2	-25.2	-18.5	-21.8	-16.8	-10.1	-10.1	-6.7	-6.7	$-\infty$

The calculation of the decoder output LLRs can then be determined using:

$$L^{(z)}(x\,|y) = \max_{\acute{s}-s\in S_1} \{A_{K-1}(\acute{s}) + \gamma_K(\acute{s},s) + B_K(s)\} - \max_{\acute{s}-s\in S_0} \{A_{K-1}(\acute{s}) + \gamma_K(\acute{s},s) + B_K(s)\}. \quad (6.6)$$

Thus the output LLRs $L^{(z)}(x\,|y)$ of the first decoder are:

$L_1{}^{(z)}(x\,|y) =$

$k/_{L^{(z)}}$	0	1	2	3	4	5	6	7	8	9
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	16.8	3.4	10.1	16.7	-10.1	6.7	10.1	-6.7	-6.7	3.4
$\boldsymbol{\beta}$	-3.4	3.4	-3.4	10.1	6.7	3.4	6.7	-3.4	-3.4	10.1
$\boldsymbol{\beta^2}$	0.0	16.8	0.0	13.4	0.0	3.4	6.7	-6.7	-6.7	6.7

	1	β^2	1	1	β	1	1	0	0	β

We observe that after making a hard decision on $L_1^{(z)}(x|y)$, there is still and error in the fourth symbol. The extrisic information can be obtained by :

$$L_e = L_1{}^{(z)}(x\,|y) - L_a^2 - L_{d,} \tag{6.7}$$

where L_a^2 is the a priori LLRs from decoder 2 that are initially set to zero in the first. This results in:

$L_e =$

$k/_{L^{(z)}}$	0	1	2	3	4	5	6	7	8	9
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	13.4	0.0	6.7	3.4	-3.4	3.4	6.7	-3.4	-3.4	6.7
$\boldsymbol{\beta}$	0.0	0.0	0.0	6.7	0.0	6.7	3.4	0.0	0.0	6.7
$\boldsymbol{\beta^2}$	0.0	10.1	0.0	6.7	0.0	3.4	0.0	0.0	0.0	6.7

The interleaved extrisic information of first decoder becomes the priori information for the second decoder, thus L_a^2 is determined as:

$L_a^2 =$

$k/_{L^{(z)}}$	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	-3.4	6.7	-3.4	0	6.7	-3.4	3.4	6.7	3.4	13.4
$\boldsymbol{\beta}$	0	6.7	0	0	0	0	6.7	3.4	6.7	0
$\boldsymbol{\beta^2}$	0	6.7	0	10.1	0	0	3.4	0	6.7	0

Now, the γ of the second decoder is obtained from the interleaved y_1 and the original received y_3:

$\gamma =$

$k/_{(s,s')}$	k	0	1	2	3	4	5	6	7	8	9
$\mathbf{0 \rightarrow 0}$		-5.3	-2.7	-8.0	-2.7	-10.7	-5.3	-5.3	-8.0	-8.0	-5.3
$\mathbf{0 \rightarrow 1}$		-5.3	-2.7	-24.0	-21.3	16.0	8.0	-13.3	-2.7	8.0	-8.0
$\mathbf{0 \rightarrow \beta}$		5.3	-10.7	-16.0	-8.0	8.0	0.0	-16.0	0.0	-13.3	-5.3
$\mathbf{0 \rightarrow \beta^2}$		-2.7	0.0	-16.0	-18.7	8.0	-5.3	-8.0	-8.0	-8.0	13.3
$\mathbf{1 \rightarrow 0}$		-2.7	-5.3	-5.3	0.0	-8.0	-8.0	-2.7	-5.3	-5.3	-8.0
$\mathbf{1 \rightarrow 1}$		-2.7	-5.3	0.0	-18.7	18.7	5.3	-10.7	0.0	10.7	-10.7
$\mathbf{1 \rightarrow \beta}$		2.7	-8.0	0.0	-10.7	5.3	2.7	-18.7	-2.7	0.0	-2.7
$\mathbf{1 \rightarrow \beta^2}$		0.0	-2.7	-18.7	-21.3	0.0	-2.7	-10.7	-10.7	-10.7	16.0
$\boldsymbol{\beta \rightarrow 0}$		-8.0	-5.3	-5.3	-5.3	-8.0	-2.7	-2.7	-5.3	-5.3	-8.0
$\boldsymbol{\beta \rightarrow 1}$		-2.7	0.0	-26.7	-18.7	13.3	5.3	0.0	-5.3	5.3	-5.3
$\boldsymbol{\beta \rightarrow \beta}$		8.0	-8.0	0.0	-5.3	5.3	-2.7	-18.7	-2.7	-16.0	-2.7
$\boldsymbol{\beta \rightarrow \beta^2}$		0.0	-8.0	-13.3	-21.3	10.7	-2.7	0.0	-5.3	-5.3	0.0
$\boldsymbol{\beta^2 \rightarrow 0}$		-5.3	-8.0	-2.7	-2.7	-5.3	-5.3	0.0	-2.7	-2.7	-10.7

$\boldsymbol{\beta^2 \to 1}$		0.0	-2.7	-24.0	0.0	16.0	2.7	-13.3	-2.7	8.0	-8.0
$\boldsymbol{\beta^2 \to \beta}$		5.3	-5.3	-21.3	-8.0	2.7	0.0	-21.3	-5.3	-18.7	0.0
$\boldsymbol{\beta^2 \to \beta^2}$		-8.0	0.0	-16.0	-24.0	8.0	0.0	-8.0	-8.0	-8.0	13.3

As before the two trace metrics *A* and *B* are:

$A =$

$k/_{\acute{S}}$	0	1	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	10
0	0	-5.33	-2.67	-10.7	-10.7	13.33	13.33	13.33	18.67	26.67	37.33
1	$-\infty$	-5.33	-2.67	-8	-5.33	5.333	21.33	18.67	16	26.67	37.33
$\boldsymbol{\beta}$	$-\infty$	5.333	0	-5.33	-10.7	16	13.33	10.67	18.67	26.67	42.67
$\boldsymbol{\beta^2}$	$-\infty$	-2.67	5.333	2.667	0	2.667	21.33	21.33	18.67	24	40

$B =$

$k/_{\acute{S}}$	0	1	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	10
0	37.33	26.67	24	32	34.67	16	13.33	18.67	5.333	-5.33	-1.39
1	26.67	24	29.33	34.67	34.67	18.67	8	8	5.333	-10.7	-1.39
$\boldsymbol{\beta}$	34.67	32	24	29.33	29.33	21.33	10.67	13.33	18.67	-2.67	-1.39
$\boldsymbol{\beta^2}$	32	24	32	34.67	37.33	13.33	16	16	10.67	13.33	-1.39

Notice that at k = 10, the metric B is now initialised to $\ln(0.25) = -1.39$ since the second encoder is not terminated, so the codeword is equally likely to end in any one of the four states. By following the same decoding procedure as we applied to the first decoder, we obtain the output LLRs of second decoder, $L_2{}^{(z)}(x\,|y)$:

$L_2{}^{(z)}(x\,|y) =$

$k/_{L^{(z)}}$	0	1	2	3	4	5	6	7	8	9
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	16.0	10.6	16.0	16.0	-21.3	26.6	26.6	-21.3	-26.6	-5.3
$\boldsymbol{\beta}$	5.3	5.3	0.0	-5.3	-16.0	0.0	-5.3	-21.3	-16.0	16.0
$\boldsymbol{\beta^2}$	10.6	16.0	0.0	5.3	-21.3	5.3	0.0	-26.6	-21.3	-10.6

Notice that all the LLR values of the non-zero elements in the fourth symbol are now negative. This means that we will obtain a zero symbol when applying a hard decision. After applying a hard decision on all of $L_2{}^{(z)}(x\,|y)$, we obtain

:

$\boldsymbol{k}$	0	1	2	3	4	5	6	7	8	9
	1	β^2	1	1	0	1	1	0	0	β

which is equal to the sum of m_1 and m_3 and shows that the turbo decoder has corrected the error.

6.5 Performance of PNC combined with Non-Binary Turbo Codes

Simulation results of PNC combined with non-binary turbo codes are now presented at the relay of a TWRC on AWGN channels and impulsive noise channels.

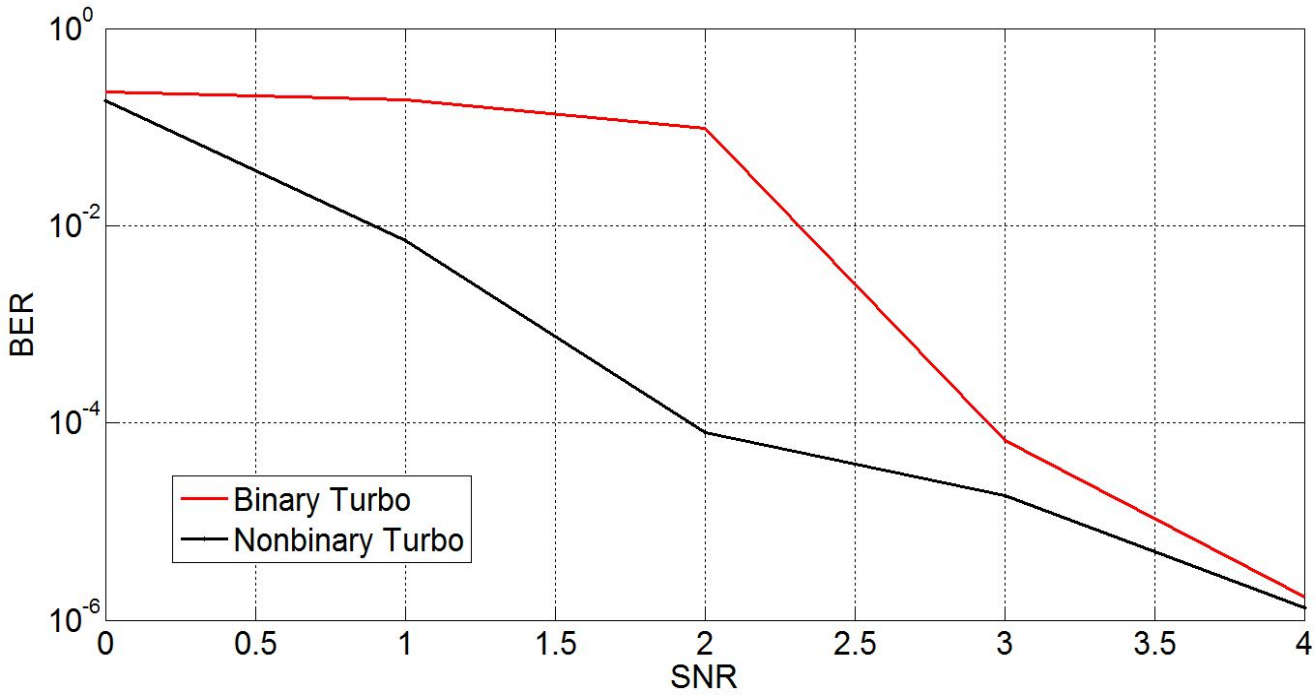


Figure 6.4 Comparison of rate $\frac{1}{2}$ $(7,5)_8$ binary turbo code and rate $\frac{1}{2}$ non-binary turbo code with β $\beta^2/1$ RSC encoders on AWGN channel.

In Figure 6.4, a comparison of rate $\frac{1}{2}$ $(7,5)_8$ binary turbo code and rate $\frac{1}{2}$ non-binary turbo code with β $\beta^2/1$ RSC convolutional encoders on AWGN channel is presented to show the performance of both coding schemes when no relay is present and act as a reference. In the figure, it can be seen that, although both codes converge at around 4dB, the non-binary turbo code has a significant advantage at the lower SNR, with approximately 1dB coding gain at BER of 10^{-4}.

Figure 6.5 shows that the advantage of non-binary turbo code remains when evaluated at the relay of a TWRC with PNC. Both codes converge at an SNR of 8dB, but the non-

binary turbo code maintains the excellent performances at low SNR, with a coding gain of approximately 1.5dB at a BER of 10^{-4}.

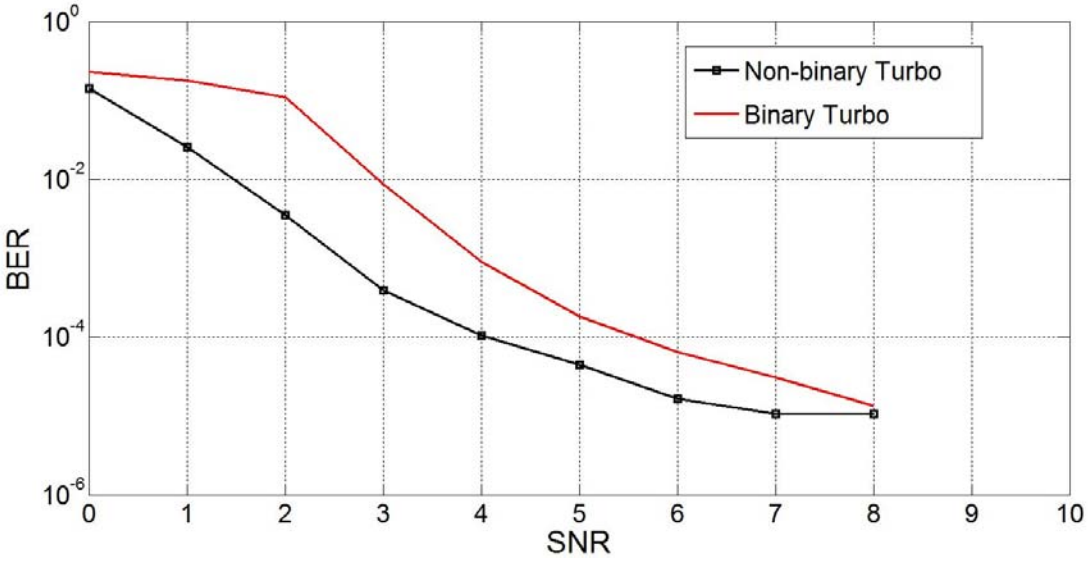


Figure 6.5 Comparison of rate $\frac{1}{2}$ $(7,5)_8$ binary turbo code and rate $\frac{1}{2}$ non-binary turbo code with β $\beta^2/1$ RSC encoders on AWGN channel decoding at the relay on PNC.

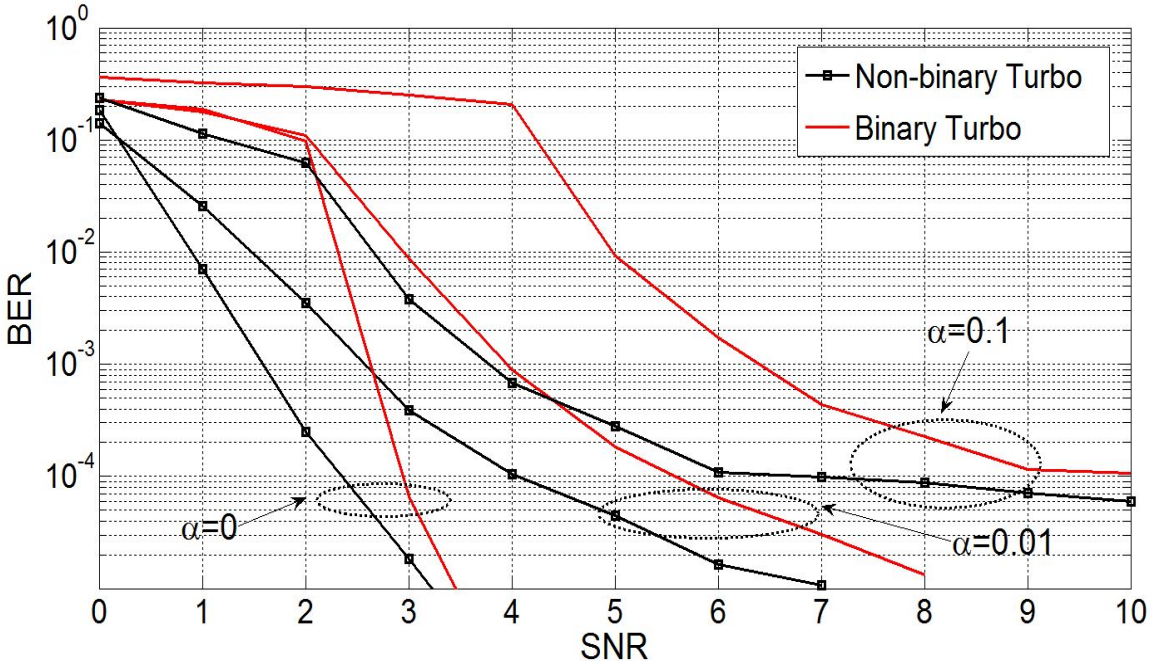


Figure 6.6 Comparison of PNC combined with rate $\frac{1}{2}$ $(7,5)_8$ binary turbo code and rate $\frac{1}{2}$ non-binary turbo code with β $\beta^2/1$ RSC encoders on different impulsive noise channels.

The comparison of PNC combined with non-binary turbo code and binary turbo codes on impulsive noise channels is shown in Figure 6.6. The channel is slightly impulsive when $\alpha = 0.01$ and the performance of both codes are seriously degraded, but the non-binary turbo code performs approximately 2dB better than binary turbo code at a BER of 10^{-4}, and this difference is even larger when the channel is more impulsive at $\alpha = 0.1$. In this situation both codes exhibit an error floor at a BER of 10^{-4} at an SNR of 6dB and 9dB respectively. However, the non-binary turbo code still has a 3dB better performance at BER of 10^{-4} compared to the binary turbo code.

6.6 Theoretical BER Analysis of PNC combined with Non-Binary Turbo Codes on Impulsive Noise Channels

In this section, upper bounds on the BER performance of turbo codes will be derived to predict the error floor and also validate the simulation results from the previous section. To derive expressions of the upper bound, we have to recall the theoretical bound for PNC combined with binary turbo codes from (4.42) in chapter 4, but multiply it by the term $\frac{2^{k-1}}{2^k-1}$ to converts symbol errors to bit errors:

$$P_{GMM} \leq \frac{2^{k-1}}{2^k-1}\left[\sum_{j=2}^{3}\frac{jn_j(1-\alpha)}{K}Q\left(\sqrt{\frac{2d_{min}RE_b}{N_G}}\right)+\sum_{j=2}^{3}\frac{\alpha jn_j}{K}Q\left(\sqrt{\frac{2d_{min}RE_b}{N_I}}\right)\right] \quad (6.10)$$

where $k = \log_2 q$ and q is the size of the finite field.

The upper bound comparison of non-binary turbo code and binary turbo code are shown in Figure 6.7 and Figure 6.8.

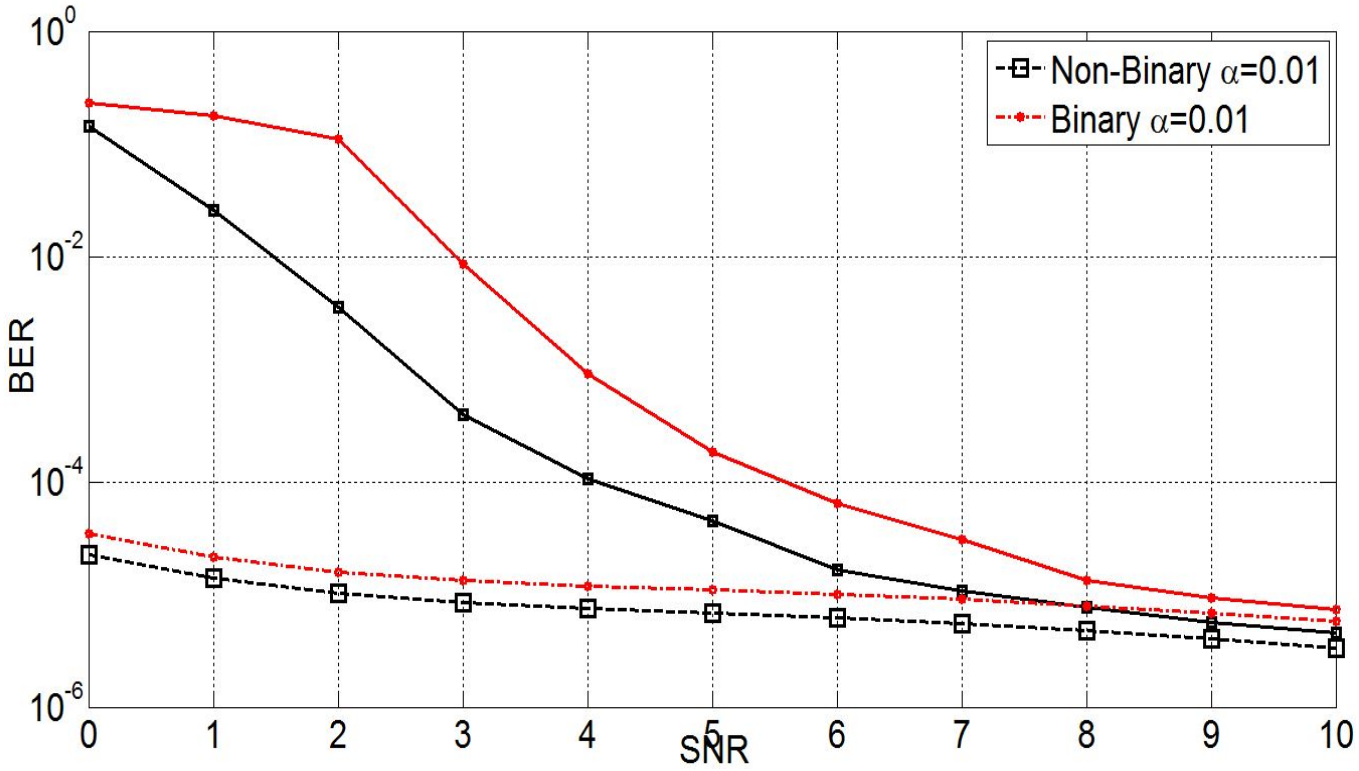


Figure 6.7 Comparison of the upper bounds on BER of PNC combined with non-binary and binary turbo codes at the relay on impulsive noise channels with $\alpha = 0.01$.

In Figure 6.7 it can be clearly seen that the upper bounds of both codes are converging with the simulation results at the high SNR region, which verifies our results. However, the non-binary turbo code consistently outperforms the binary turbo code, particularly at low SNRs.

Figure 6.8 shows the upper bounds on BER of PNC combined with both codes at the relay on impulsive noise channels when $\alpha = 0.01$ and 0.1. From the comparison it can be concluded that the non-binary turbo code outperforms the binary turbo code at all SNRs and non-binary turbo codes have a lower error floor than the binary turbo codes at all times.

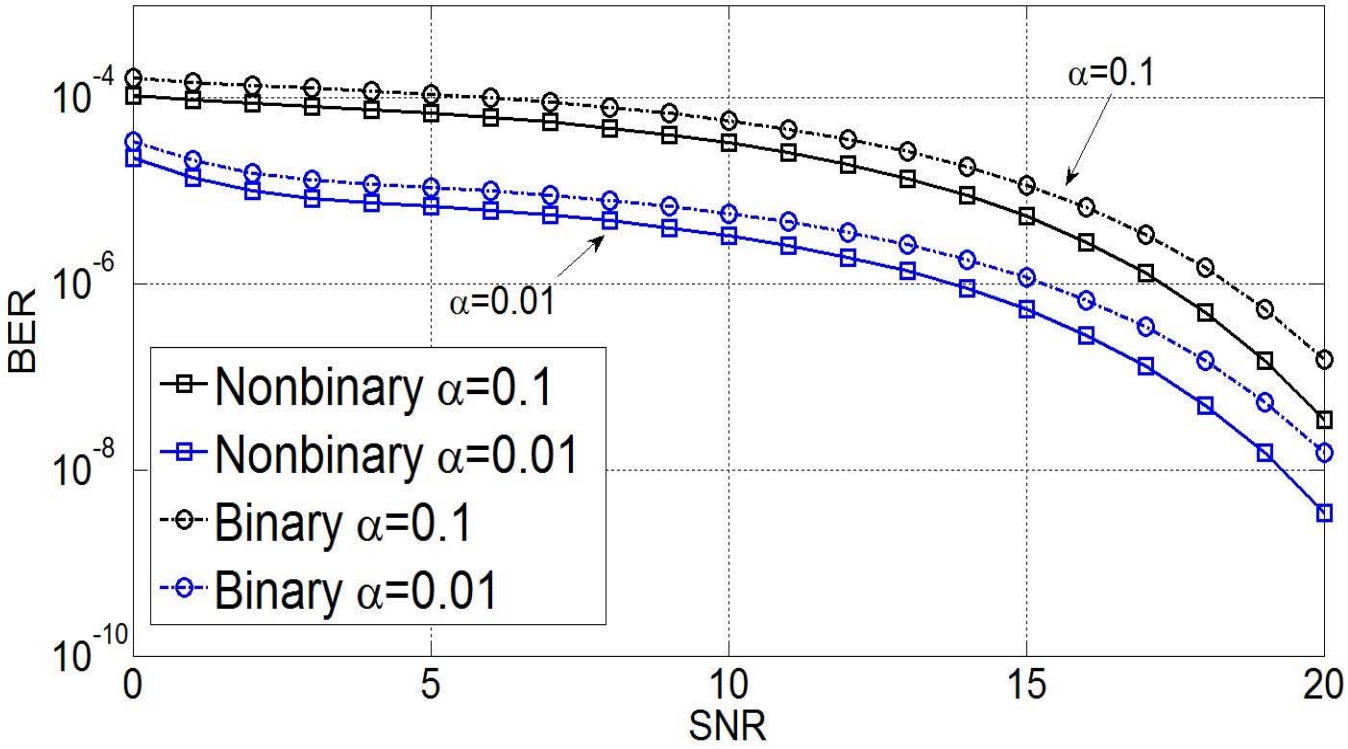


Figure 6.8 Lower bound on BER of PNC combined with non-binary and binary turbo codes at the relay on impulsive noise channels when $\alpha = 0.01$ and 0.1.

We might assume that the codes defined in larger finite fields with more states should achieve better performance than the codes defined in the small finite fields, due to the increase in minimum Hamming distance. However, we have observed that the performance of non-binary turbo codes peaks when the number of states is around 8, but for higher number of states the performance of the codes degrades. A theoretical upper bound comparison on the BER of non-binary turbo codes defined in GF(2), GF(4) and GF(16) highlights the difference in performance, as shown in Figure 6.9 and Figure 6.10. The free distances of turbo codes defined in different GF(q) are giving in table 6.1.

Field	GF(2)	GF(4)	GF(16)
d_{free}	5	5	3

Table 6.1 The free distance of turbo codes defined in different GF(q)

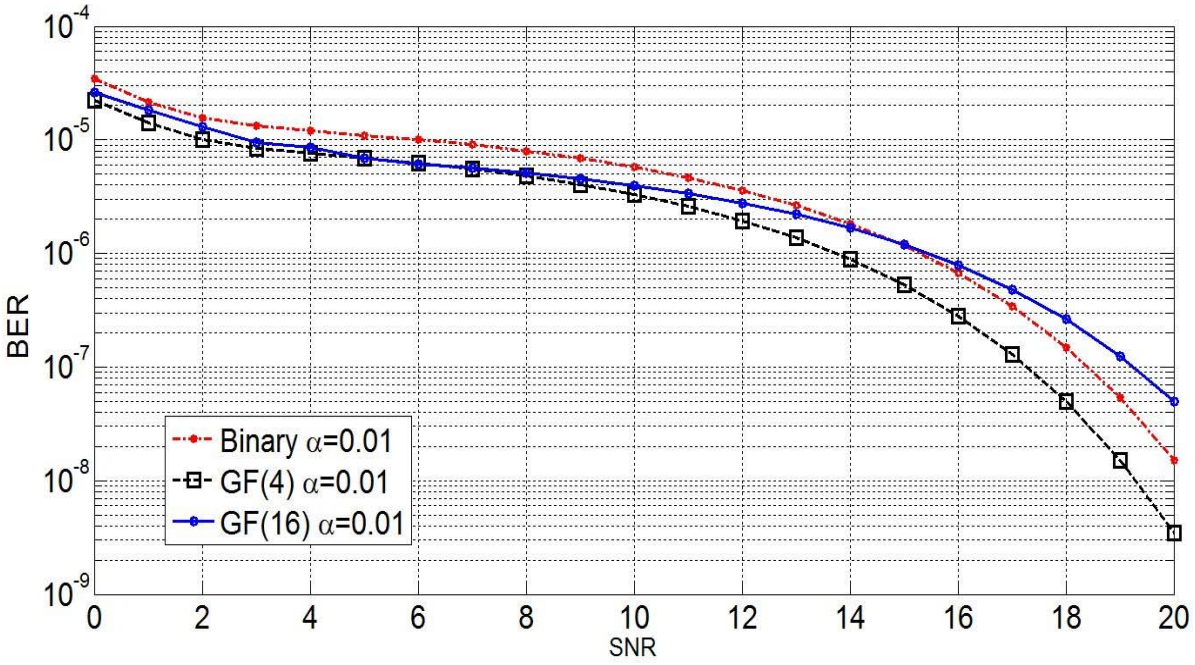


Figure 6.9 Lower bound of GF(4) and GF(16) non-binary turbo codes and binary turbo code decoding at the relay on PNC under impulsive channel when $\alpha = 0.01$.

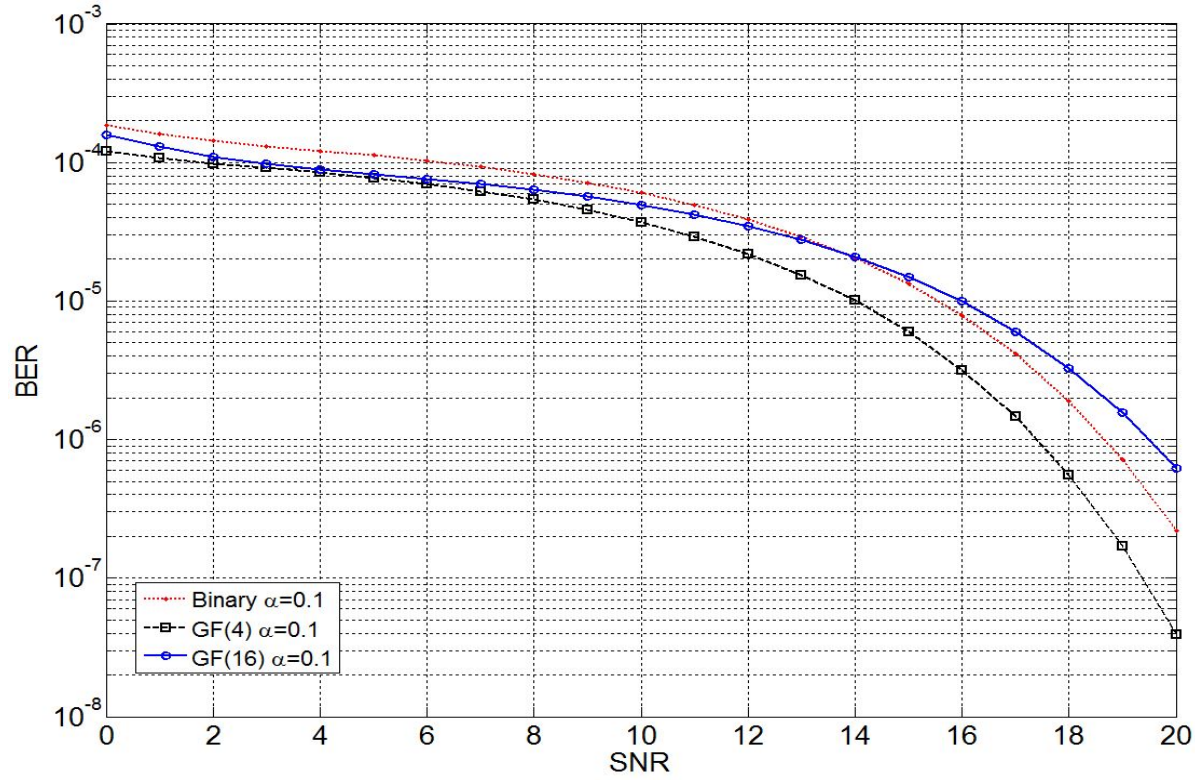


Figure 6.10 Lower bound of GF(4) and GF(16) non-binary turbo codes and binary turbo code decoding at the relay on PNC under impulsive channel when $\alpha = 0.1$.

6.7 Conclusions

In this chapter, an analysis of non-binary turbo codes on a TWRC employing PNC with additive impulsive noise channels has been investigated. The decoding procedures of non-binary turbo decoders, including a complete worked example of one single turbo decoder iteration, are detailed. We have shown that the performance of turbo codes is severely affected on the GMM noise channel when the mixture is high, but that non-binary turbo codes achieve significant improvements in performance at low SNR compared with binary turbo codes. Finally, an upper bound on BER to determine the error floor in the presence of impulsive noise was derived for both binary and non-binary turbo codes to validate our simulation results and was observed to be consistently lower for non-binary turbo codes for a broad range of SNRs. We have shown that the performance of non-binary turbo codes does not keep increasing as finite field size increases, with higher error floors observed for a non-binary turbo code defined in GF(16).

Chapter 7

Conclusions and Further Work

7.1 Conclusions

This thesis has focused on the performance of PNC combined with non-binary coding schemes on impulsive noise channels. After a comprehensive literature survey, it was found that this is a research topic that has not been considered by the academic community despite evidence that non-binary codes can outperform binary codes on harsher channels. Furthermore, wireless networks can be deployed in urban environments where man-made interference can cause impulses to be added to the transmitted signals. Hence, if PNC is employed at the relays of a wireless network in this environment, it is very important to have a good understanding of how the performance will be affected and also find solutions that can mitigate the effects of the channel. This was the major motivation for the thesis and interesting and original results have been achieved that show the advantages of non-binary codes when combined with PNC on impulsive noise channels.

In chapter 4, the combination of binary turbo codes and trellis BICM-ID with PNC was investigated on AWGN and additive impulsive noise channels modelled using the Gaussian mixture model. On the AWGN channel, the turbo codes achieved a superior performance, but interestingly it was observed that trellis BICM-ID outperformed turbo codes when the channel became more impulsive. This was due to the bit interleaver increasing the code diversity and also the simplified decoding scheme that prevents errors from propagating in each iteration. Therefore, it has been shown that trellis BICM-ID is a better choice of coding scheme to be combined with PNC when the channels are impulsive, since the performance is better than turbo codes and there is also a reduction in complexity.

In chapter 5, non-binary convolutional codes were combined with PNC on impulsive noise channels. This is original work and presented the first results of non-binary convolutional codes combined with PNC. To make a fair comparison between binary and non-binary convolutional codes, the number of states was kept fixed at 256. On the AWGN channel, the convolutional codes defined in GF(2), GF(4) and GF(16) all performed the same as expected. However, on impulsive noise channels it was observed that the non-binary convolutional code defined in GF(4) outperformed the binary convolutional code, but the non-binary convolutional code defined in GF(16) performed worse. This is due to the shorter constraint length of the GF(16) code, which results in the

minimum length of a path diverging from and converging to the all-zero path being shorter.

In chapter 6, the first results of PNC combined with non-binary turbo codes were presented. It was shown that the non-binary turbo codes outperform binary turbo codes at low SNRs for both AWGN and impulsive noise channels, but then the performances converged at higher SNRs. As the impulsiveness increased, the coding gain at low SNRs became more significant. Furthermore, the error floor in the performance due to impulsive noise was consistently lower for non-binary turbo codes. However, increasing the finite field size did not result in further gains in performance. The non-binary turbo code defined in GF(16) performed worse despite having a higher finite field and more states. This is most likely due to the higher number of states, since it is known that binary turbo code performance peaks when the number of states is around 8. When the number of states is higher, the BER performance of the convolutional code has a steeper waterfall region, but has a worse BER performance at low SNRs. However, it is the performance at low SNRs that determines the performance of the turbo code. This argument could also be applied to non-binary turbo codes, but requires further investigation.

7.2 Further work

As we have presented the application of channel coding schemes to PNC over impulsive channels, it opens new and challenging aims for the future. In the binary coding area, we have shown that the iterative decoder can fail to correct the errors caused by impulsive noise, since the LLRs of the error bits are too large making the decoder believe that certain bits are very reliable. Thus, a simpler iterative decoder for impulsive channel is necessary, such as that used in trellis BICM-ID, since it is important to keep the complexity at the relay as low as possible. Second, the best modulation schemes for channel coded PNC need to be ensured for different channel coding schemes. In this thesis we are using QPSK modulation scheme for all codes, resulting in a 9-QAM constellation at the relay. However, higher level of modulation schemes could increase the transmission capacity but at a cost of higher complexity at the relay, e.g. 8-PSK modulated signals at node results in 17-QAM at the relay. This may increase the difficulty of demapping the signals as well as the theoretical analysis of the system. Thus,

suitable modulation schemes for PNC in realistic situations need to be considered. Third, the design of non-binary codes defined in larger GF(q) is an interesting topic. We have shown that the non-binary codes have a significant improvement of BER at lower SNR, especially for impulsive channels. Furthermore, the codes defined in larger GF(q) have more capability to correct errors since there are more bits contained in a single symbol. Therefore, finding the optimal codes defined in larger GF(q) is critical to optimise the performance of PNC on more realistic channels.

Reference

[1] Ojanpera T, Prasad R. An overview of air interface multiple access for IMT-2000/UMTS[J]. Communications Magazine, IEEE, 1998, 36(9): 82-86, 91-5.

[2] Li J, Blake C, De Couto D S J, et al. Capacity of ad hoc wireless networks[C]//Proceedings of the 7th annual international conference on Mobile computing and networking. ACM, 2001: 61-69.

[3] Ng P C, Liew S C. Offered load control in IEEE 802.11 multi-hop ad-hoc networks[C]//Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on. IEEE, 2004: 80-89.

[4] Zhang, S., Liew, S.C. and Lam, P.P., 2006, September. Hot topic: physical-layer network coding. In *Proceedings of the 12th annual international conference on Mobile computing and networking* (pp. 358-365). ACM.

[5] Zhang, S., Liew, S.C. and Lam, P.P., 2006, October. On the synchronization of physical-layer network coding. In *Information Theory Workshop, 2006. ITW'06 Chengdu. IEEE* (pp. 404-408). IEEE.

[6] Katti, S., Rahul, H., Hu, W., Katabi, D., Médard, M. and Crowcroft, J., 2006, September. XORs in the air: practical wireless network coding. In *ACM SIGCOMM computer communication review* (Vol. 36, No. 4, pp. 243-254). ACM.

[7] Hausl, C. and Dupraz, P., 2006, September. Joint network-channel coding for the multiple-access relay channel. In *Sensor and Ad Hoc Communications and Networks, 2006. SECON'06. 2006 3rd Annual IEEE Communications Society on* (Vol. 3, pp. 817-822). IEEE.

[8] Popovski, P. and Yomo, H., 2007, June. Physical network coding in two-way wireless relay channels. In *Communications, 2007. ICC'07. IEEE International Conference on* (pp. 707-712). IEEE.

[9] Partan, J., Kurose, J. and Levine, B.N., 2007. A survey of practical issues in underwater networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, *11*(4), pp.23-33.

[10] Zhang, S., Liew, S.C. and Lu, L., 2008, November. Physical layer network coding schemes over finite and infinite fields. In *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE* (pp. 1-6). IEEE.

[11] Katti, S., Katabi, D., Balakrishnan, H. and Medard, M., 2008, August. Symbol-level network coding for wireless mesh networks. In *ACM SIGCOMM Computer Communication Review* (Vol. 38, No. 4, pp. 401-412). ACM.

[12] Zhang, S. and Liew, S.C., 2009. Channel coding and decoding in a relay system operated with physical-layer network coding. *Selected Areas in Communications, IEEE Journal on*, *27*(5), pp.788-796.

[13] Rossetto, F. and Zorzi, M., 2009, June. On the design of practical asynchronous physical layer network coding. In *IEEE 10th Workshop on SPAWC* (Vol. 9, pp. 469-473).

[14] Koike-Akino, T., Popovski, P. and Tarokh, V., 2009. Optimized constellations for two-way wireless relaying with physical network coding. *Selected Areas in Communications, IEEE Journal on*, *27*(5), pp.773-787.

[15] Cui, T., Gao, F. and Tellambura, C., 2009. Differential modulation for two-way wireless communications: a perspective of differential network coding at the physical layer. *Communications, IEEE Transactions on*, *57*(10), pp.2977-2987.

[17] Louie R H Y, Li Y, Vucetic B. Practical physical layer network coding for two-way relay channels: performance analysis and comparison[J]. Wireless Communications, IEEE Transactions on, 2010, 9(2): 764-777.

[18] Nazer B, Gastpar M. Reliable physical layer network coding[J]. Proceedings of the IEEE, 2011, 99(3): 438-460.

[19] Wilson M P, Narayanan K, Pfister H D, et al. Joint physical layer coding and network coding for bidirectional relaying[J]. Information Theory, IEEE Transactions on, 2010, 56(11): 5641-5654.

[20] Lu L, Liew S C. Asynchronous physical-layer network coding[J]. Wireless Communications, IEEE Transactions on, 2012, 11(2): 819-831.

[21] Wübben D, Lang Y. Generalized sum-product algorithm for joint channel decoding and physical-layer network coding in two-way relay systems[C]//Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE. IEEE, 2010: 1-5.

[22] Ding Z, Krikidis I, Thompson J, et al. Physical layer network coding and precoding for the two-way relay channel in cellular systems[J]. Signal Processing, IEEE Transactions on, 2011, 59(2): 696-712.

[23] To D, Choi J. Convolutional codes in two-way relay networks with physical-layer network coding[J]. Wireless Communications, IEEE Transactions on, 2010, 9(9): 2724-2729.

[24] Lu L, Liew S C, Zhang S. Optimal decoding algorithm for asynchronous physical-layer network coding[C]//Communications (ICC), 2011 IEEE International Conference on. IEEE, 2011: 1-6.

[25] Ju M C, Kim I M. Error performance analysis of BPSK modulation in physical-layer network-coded bidirectional relay networks[J]. Communications, IEEE Transactions on, 2010, 58(10): 2770-2775.

[26] Khan M N, Ghauri S. The WiMAX 802.16 e physical layer model[C]//Wireless, Mobile and Multimedia Networks, 2008. IET International Conference on. IET, 2008: 117-120.

[27] Zhan A, He C. Joint design of channel coding and physical network coding for wireless networks[C]//Neural Networks and Signal Processing, 2008 International Conference on. IEEE, 2008: 512-516.

[28] Wang D, Fu S, Lu K. Channel coding design to support asynchronous physical layer network coding[C]//Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE. IEEE, 2009: 1-6.

[29] Gacanin H, Adachi F. Performance of physical layer network coding in a frequency-selective fading channel[C]//Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on. IEEE, 2009: 928-932.

[30] To D, Choi J. Convolutional codes in two-way relay networks with physical-layer network coding[J]. Wireless Communications, IEEE Transactions on, 2010, 9(9): 2724-2729.

[31] Lu L, Wang T, Liew S C, et al. Implementation of physical-layer network coding[J]. Physical Communication, 2013, 6: 74-87.

[32] Yang Q, Liew S C. Asynchronous convolutional-coded physical-layer network coding[J]. Wireless Communications, IEEE Transactions on, 2015, 14(3): 1380-1395.

[33] Hausl, C. and Hagenauer, J., 2006, June. Iterative network and channel decoding for the two-way relay channel. In *Communications, 2006. ICC'06. IEEE International Conference on* (Vol. 4, pp. 1568-1573). IEEE.

[34] Zhang, S., Zhu, Y., Liew, S.C. and Ben Letaief, K., 2007, March. Joint design of network coding and channel decoding for wireless networks. In *Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE* (pp. 779-784). IEEE.

[35] Fang, D. and Burr, A., 2011, September. Performance Degradation of Turbo Coded Physical Layer Network Coding on the Two-Way Relay Channel. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2011 IEEE 22nd International Symposium on* (pp. 1748-1752). IEEE.

[36] Ke, G. and Jing, W., 2010, April. Improvement of physical-layer network coding method in relay networks. In *Communications and Mobile Computing (CMC), 2010 International Conference on* (Vol. 3, pp. 457-460). IEEE.

[37] Zeng, W., Wu, X. and Yang, Z., 2013, April. Turbo DPSK in Bi-directional relaying. In *Wireless Communications and Networking Conference Workshops (WCNCW), 2013 IEEE* (pp. 155-159). IEEE.

[38] Li X, Zhang S, Qian G. Mapping and coding design for channel coded physical-layer network coding[C]//High Mobility Wireless Communications (HMWC), 2013 International Workshop on. IEEE, 2013: 120-125.

[39] M. O. Hasna and M. S. Alouini, "End-to-end performance of transmission systems with relays over Rayleigh-fading channels," *IEEE Trans. Wireless Commun.*, vol. 2, no. 6, pp. 1126–1131, Nov. 2003.

[40] Noori M, Ardakani M. On symbol mapping for binary physical-layer network coding with PSK modulation[J]. Wireless Communications, IEEE Transactions on, 2012, 11(1): 21-26.

[41] Huang T, Yang T, Yuan J, et al. Convergence analysis for channel-coded physical layer network coding in Gaussian two-way relay channels[C]//Wireless Communication Systems (ISWCS), 2011 8th International Symposium on. IEEE, 2011: 849-853.
[42] P. A. Anghel, "Exact symbol error probability of a cooperative network in a Rayleigh-fading environment," *IEEE Trans. Wireless Commun.*, vol. 3, no. 5, pp. 1416–1421, Sep. 2004.
[43] Goodman N R. Statistical analysis based on a certain multivariate complex Gaussian distribution (an introduction)[J]. The Annals of mathematical statistics, 1963, 34(1): 152-177.
[44] Granger C W J. Prediction with a generalized cost of error function[J]. OR, 1969: 199-207.

[45] Padgett J E, Günther C G, Hattori T. Overview of wireless personal communications[J]. Communications Magazine, IEEE, 1995, 33(1): 28-41.

[46] Pinsker M S. The probability of error in block transmission in a memoryless Gaussian channel with feedback[J]. Problemy Peredachi Informatsii, 1968, 4(4): 3-19.

[47] Von Helmholtz H. Handbuch der physiologischen Optik[M]. Voss, 1867.

[48] Rider P R. Generalized cauchy distributions[J]. Annals of the Institute of Statistical mathematics, 1957, 9(1): 215-223.

[49] Lévy P. Les lois de probabilité dans les ensembles abstraits[J]. Revue de Métaphysique et de Morale, 1925, 32(2): 149-174.

[50] Tukey J W. A survey of sampling from contaminated distributions[J]. Contributions to probability and statistics, 1960, 2: 448-485.

[51] Zivkovic Z. Improved adaptive Gaussian mixture model for background subtraction[C]//Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on. IEEE, 2004, 2: 28-31.

[52] Rasmussen C E. The Infinite Gaussian Mixture Model[C]//NIPS. 1999, 12: 554-560.

[53] Reynolds D A, Quatieri T F, Dunn R B. Speaker verification using adapted Gaussian mixture models[J]. Digital signal processing, 2000, 10(1): 19-41.

[54] Yang M H, Ahuja N. Gaussian mixture model for human skin color and its applications in image and video databases[C]//Electronic Imaging'99. International Society for Optics and Photonics, 1998: 458-466.

[55] Huang Y, Englehart K B, Hudgins B, et al. A Gaussian mixture model based classification scheme for myoelectric control of powered upper limb prostheses[J]. Biomedical Engineering, IEEE Transactions on, 2005, 52(11): 1801-1811.

[56] KaewTraKulPong P, Bowden R. An improved adaptive background mixture model for real-time tracking with shadow detection[M]//Video-based surveillance systems. Springer US, 2002: 135-144.

[57] Xuan G, Zhang W, Chai P. EM algorithms of Gaussian mixture model and hidden Markov model[C]//Image Processing, 2001. Proceedings. 2001 International Conference on. IEEE, 2001, 1: 145-148.

[58] Berrou, C., Glavieux, A., & Thitimajshima, P. (1993). Near Shannon limit error-correcting coding and decoding.

[59] Berrou C, Glavieux A. Near optimum error correcting coding and decoding: Turbo-codes[J]. Communications, IEEE Transactions on, 1996, 44(10): 1261-1271.

[60] Ungerboeck, G., 1982. Channel coding with multilevel/phase signals. *Information Theory, IEEE Transactions on*, *28*(1), pp.55-67.

[61] Zehavi, E., 1992. 8-PSK trellis codes for a Rayleigh channel. *Communications, IEEE Transactions on*, *40*(5), pp.873-884.

[62] Caire, G., Taricco, G. and Biglieri, E., 1998. Bit-interleaved coded modulation. *Information Theory, IEEE Transactions on*, *44*(3), pp.927-946.

[63] Li, X. and Ritcey, J.A., 1997. Bit-interleaved coded modulation with iterative decoding. *Communications Letters, IEEE*, *1*(6), pp.169-171.

[64] Li, X. and Ritcey, J.A., 1999. Bit-interleaved coded modulation with iterative decoding. In *Communications, 1999. ICC'99. 1999 IEEE International Conference on* (Vol. 2, pp. 858-863). IEEE.

[65] Elias P. Coding for two noisy channels[C]//Information Theory, Third London Symposium. 1955, 67.

[66] Viterbi A J. Two constructive classes of convolutional codes for multiple-signal channels[C]//Presented at the IEEE Int. Symp. Inform. Theory. 1972.

[67] Viterbi A J, Jacobs I M. Advances in coding and modulation for noncoherent channels affected by fading, partial band, and multiple-access interference[C]//In: Advances in communication systems: Theory and applications. Volume 4.(A76-35851 17-32) New York, Academic Press, Inc., 1975, p. 279-308. 1975, 4: 279-308.

[68] Blokh È L, Zyablov V V. Coding of generalized concatenated codes[J]. Problemy Peredachi Informatsii, 1974, 10(3): 45-50.

[69] Bossert M. Decoding of generalized concatenated codes[M]//Applied Algebra, Algebraic Algorithms and Error-Correcting Codes. Springer Berlin Heidelberg, 1988: 89-98.

[70] Odenwalder U P. Dual-k convolutional codes for noncoherently demodulated channels[C]//International Telemetering Conference. 1976, 1: 165-174.

[71] Trumpis B D. Convolutional coding for M-ary channels[M]. 1975.

[72] Ouahada K. Nonbinary convolutional codes and modified M-FSK detectors for power-line communications channel[J]. Communications and Networks, Journal of, 2014, 16(3): 270-279.

[73] Boreli R, Yates K W. Convolutional coding for a phase modulated FCMA system[C]//Spread Spectrum Techniques and Applications, 1994. IEEE ISSSTA'94., IEEE Third International Symposium on. IEEE, 1994: 273-276.

[74] Konishi T. A coded modulation scheme for 64-QAM with a matched mapping[C]//Information Theory and its Applications (ISITA), 2014 International Symposium on. IEEE, 2014: 191-195.

[74] Ouahada K. Nonbinary convolutional codes and modified M-FSK detectors for power-line communications channel[J]. Communications and Networks, Journal of, 2014, 16(3): 270-279.

[76] Trumpis B D. Convolutional coding for M-ary channels[M]. 1975.

[77] Faraji-Dana Z, Mitran P. On non-binary constellations for channel-coded physical-layer network coding[J]. Wireless Communications, IEEE Transactions on, 2013, 12(1): 312-319.

[78] Hausl, C. and Philippe, D., 2006. Joint Network Channel Coding for the Multiple Access Relay Channel. 3rd annual IEEE conference on Sensors and Ad Hoc Communications and Networks (pp. 817-822). IEEE.

[79] Fang, D. and Burr, A., 2011, September. Performance Degradation of Turbo Coded Physical Layer Network Coding on the Two-Way Relay Channel. In Personal Indoor and Mobile Radio Communications (PIMRC), 2011 IEEE 22nd International Symposium on (pp. 1748-1752).

[80] Berrou, C. and Jezequel, M., 1999. Non-binary convolutional codes for turbo coding. Electronics Letters, 35(1), pp.39-40.

[81] Middleton, D., 1979. Procedures for determining the parameters of the first-order canonical models of Class A and Class B electromagnetic interference. Electromagnetic Compatibility, IEEE Transactions on, (3), pp.190-208.

[82]Zhang S, Liew S C, Lam P P. Hot topic: physical-layer network coding[C]//Proceedings of the 12th annual international conference on Mobile computing and networking. ACM, 2006: 358-365.

[83] Chen W, Letaief K, Cao Z. Opportunistic network coding for wireless networks[C]//Communications, 2007. ICC'07. IEEE International Conference on. IEEE, 2007: 4634-4639.

[84] Zhan A, He C. Joint design of channel coding and physical network coding for wireless networks[C]//Neural Networks and Signal Processing, 2008 International Conference on. IEEE, 2008: 512-516.

[85] Liu P, Kim I M. Outage probability analysis of physical-layer network coding in bidirectional relay networks[C]//2010 25th Biennial Symposium on Communications. 2010.

[86]Hu P, Ibnkahla M. A survey of physical-layer network coding in wireless networks[C]//Communications (QBSC), 2010 25th Biennial Symposium on. IEEE, 2010: 311-314.

[87] Chen C, Xiang H. The throughput order of ad hoc networks with physical-layer network coding and analog network coding[C]//Communications, 2008. ICC'08. IEEE International Conference on. IEEE, 2008: 2146-2152.

[88] Huang J, Wang Z, Zhou S, et al. Turbo equalization for OFDM modulated physical layer network coding[C]//Signal Processing Advances in Wireless Communications (SPAWC), 2011 IEEE 12th International Workshop on. IEEE, 2011: 291-295.

[89] Liang Y C, Zhang R. Optimal analogue relaying with multi-antennas for physical layer network coding[C]//Communications, 2008. ICC'08. IEEE International Conference on. IEEE, 2008: 3893-3897.

[90] Lu K, Fu S, Qian Y, et al. On capacity of random wireless networks with physical-layer network coding[J]. Selected Areas in Communications, IEEE Journal on, 2009, 27(5): 763-772.

[91] Zhang S, Liew S C. Physical layer network coding with multiple antennas[C]//Wireless Communications and Networking Conference (WCNC), 2010 IEEE. IEEE, 2010: 1-6.

[92] Blackard K L, Rappaport T S, Bostian C W. Measurements and models of radio frequency impulsive noise for indoor wireless communications[J]. Selected Areas in Communications, IEEE Journal on, 1993, 11(7): 991-1001.

[93] Blankenship T K, Rappaport T S. Characteristics of impulsive noise in the 450-MHz band in hospitals and clinics[J]. Antennas and Propagation, IEEE Transactions on, 1998, 46(2): 194-203.

[94] Kotaki M. Global distribution of atmospheric radio noise derived from thunderstorm activity[J]. Journal of atmospheric and terrestrial physics, 1984, 46(10): 867-877.

[95] Bahl L R, Cocke J, Jelinek F, et al. 284 IEEE TRANSACTTONS ON INFORMATION THEORY, MARCH 1974[J]. 1974.

[96] Berrou C. Near Shannon Limit Error-Correction Code: Turbo Code[C]//Proceedings of the 1993 IEEE International Conference on Communications. 1993.

[97] Ritcey J A. Bit-interleaved coded modulation with iterative decoding using soft feedback[J]. Electronics Letters, 1998, 34(10): 942-943.

[98] Chitre M, Shahabudeen S, Stojanovic M. Underwater acoustic communications and networking: Recent advances and future challenges[J]. Marine technology society journal, 2008, 42(1): 103-116.

[99] Yousuf M S, Rizvi S Z, El-Shafei M. Power line communications: an overview-Part II[C]//Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on. IEEE, 2008: 1-6.

[100] Bahl L R, Cocke J, Jelinek F, et al. 284 IEEE TRANSACTTONS ON INFORMATION THEORY, MARCH 1974[J]. 1974.

[101] Dholakia A. Majority-Logic Decoding[M]//Introduction to Convolutional Codes with Applications. Springer US, 1994: 127-136.

[102] Wozencraft J M. Sequential decoding for reliable communication[J]. 1957.

[103] Lin S, Costello D J. Error control coding[M]. Pearson Education India, 2004.

[104] L.Bahl, J.Cocke, F.Jelinek, and J.Raviv, "Optimal Decoding of Linear Codes for minimizing symbol error rate", IEEE Transactions on Information Theory, vol. IT-20(2), pp. 284-287, March 1974.

[105] Wang S, Patenaude F. A systematic approach to modified BCJR MAP algorithms for convolutional codes[J]. EURASIP Journal on Advances in Signal Processing, 2006, 2006(1): 1-15.

[106] Robertson P, Hoeher P, Villebrun E. Optimal and sub‐optimal maximum a posteriori algorithms suitable for turbo decoding[J]. European Transactions on Telecommunications, 1997, 8(2): 119-125.

[107] Wicker S B. Error control systems for digital communication and storage[M]. Englewood Cliffs: Prentice hall, 1995.

[108] Proakis J G. Digital communications. 1995[J]. McGraw-Hill, New York.

Appendix

1. In appendix 1 its showing the derivation of simplifies the LLR of turbo decoder $L(y|m_R^1)$ and $L(y|m_R^2)$ from (4.16) to (4.17).

   From the resulting nine-point constellation diagram at the relay, each received symbol y is demapped to a pair of LLR values, $L(y|m_R^1)$ and $L(y|m_R^2)$, which are a measure of the reliability of the two XORed transmitted bits, m_R^1 and m_R^2.

   From (4.14) and (4.15) we have

$$P(y|m_R^1 = 0) = P(y|m_R = 00) + P(y|m_R = 01)$$
$$= \sqrt{\frac{1}{2\pi\sigma^2}}\left(e^{-\frac{(y_Q-2E)^2}{2\sigma^2}} + e^{-\frac{(y_Q+2E)^2}{2\sigma^2}}\right)\left(e^{-\frac{(y_I-2E)^2}{2\sigma^2}} + e^{-\frac{(y_I+2E)^2}{2\sigma^2}} + 2e^{-\frac{(y_I)^2}{2\sigma^2}}\right) \quad (4.14)$$

$$P(y|m_R^1 = 1) = P(y|m_R = 10) + P(y|m_R = 11)$$
$$= 2\sqrt{\frac{1}{2\pi\sigma^2}}e^{-\frac{(y_Q)^2}{2\sigma^2}}\left(e^{-\frac{(y_I-2E)^2}{2\sigma^2}} + e^{-\frac{(y_I+2E)^2}{2\sigma^2}} + 2e^{-\frac{(y_I)^2}{2\sigma^2}}\right) \quad (4.15)$$

   Therefore, by submitting (4.14) and (4.15) into (4.13) the reliability of y can be rewritten as:

$$L(y|m_R^1) = \ln\left(\frac{P(y|m_R = 10) + P(y|m_R = 11)}{P(y|m_R = 00) + P(y|m_R = 01)}\right)$$

$$= \ln\left[\frac{\left(e^{-\frac{(y_Q-2E)^2}{2\sigma^2}} + e^{-\frac{(y_Q+2E)^2}{2\sigma^2}}\right)\left(e^{-\frac{(y_I-2E)^2}{2\sigma^2}} + e^{-\frac{(y_I+2E)^2}{2\sigma^2}} + 2e^{-\frac{(y_I)^2}{2\sigma^2}}\right)}{2e^{-\frac{(y_Q)^2}{2\sigma^2}}\left(e^{-\frac{(y_I-2E)^2}{2\sigma^2}} + e^{-\frac{(y_I+2E)^2}{2\sigma^2}} + 2e^{-\frac{(y_I)^2}{2\sigma^2}}\right)}\right]$$

$$= \ln\left[\frac{\left(e^{-\frac{\left(y_Q-2E\right)^2}{2\sigma^2}} + e^{-\frac{\left(y_Q+2E\right)^2}{2\sigma^2}}\right)}{2e^{-\frac{\left(y_Q\right)^2}{2\sigma^2}}}\right]$$

$$= \ln\left[\frac{e^{-\frac{2y_Q}{\sigma^2}} + e^{\frac{2y_Q}{\sigma^2}}}{2e^{\frac{1}{\sigma^2}}}\right] = \ln\left(\cosh\frac{2Ey_Q}{\sigma^2}\right) - \frac{1}{\sigma^2}.$$

2. In appendix 2 we are giving the full extension expression of the LLR exchanged between two turbo decoders with the *a priori* LLR.

   Extension of (4.21) and (4.22)

   The full expression of (4.21) is:

$$L(m_R^1|y)$$
$$= L_a(m_R^1)$$
$$+ \ln\left(\frac{4e^{-\frac{(y_I^2+y_Q^2)}{2\sigma^2}} + 2e^{-\frac{\left[(y_I-2E)^2+(y_Q)^2\right]}{2\sigma^2}} + 2e^{-\frac{\left[(y_I+2E)^2+(y_Q)^2\right]}{2\sigma^2}} e^{L_a(m_R^2)}}{e^{-\frac{\left[(y_I-2E)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I-2E)^2+(y_Q+2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I+2E)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I+2E)^2+(y_Q+2E)^2\right]}{2\sigma^2}} + 2\gamma e^{-\frac{\left[(y_I)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + 2\gamma e^{-\frac{\left[(y_I)^2+(y_Q+2E\right.}{2\sigma^2}}}\right)$$

   And the full expression of (4.22) is:

$$L(m_R^2|y)$$
$$= L_a(m_R^2)$$
$$+ \ln\left(\frac{2e^{-\frac{\left[(y_I)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + 2e^{-\frac{\left[(y_I)^2+(y_Q+2E)^2\right]}{2\sigma^2}} + \left(2e^{-\frac{\left[(y_I-2E)^2+(y_Q)^2\right]}{2\sigma^2}} + 2e^{-\frac{\left[(y_I+2E)^2+(y_Q)^2\right]}{2\sigma^2}}\right) e^{L_a(m_R^1)}}{e^{-\frac{\left[(y_I-2E)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I-2E)^2+(y_Q+2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I+2E)^2+(y_Q-2E)^2\right]}{2\sigma^2}} + e^{-\frac{\left[(y_I+2E)^2+(y_Q+2E)^2\right]}{2\sigma^2}} + 4e^{-\frac{\left(y_I^2+y_Q^2\right)}{2\sigma^2}} e\right)$$

e